

YILDIZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

128644

ELEKTRONİK DOKÜMAN YÖNETİM SİSTEMİ

TC. YÜKSEKÖĞRETİM KURULU
DOKÜMANTASYON MERKEZİ

Bilgisayar Müh. İtri Osman TUFANOĞULLARI

F.B.E. Bilgisayar Mühendisliği Anabilim Dalında Hazırlanan

YÜKSEK LİSANS TEZİ

Tez Danışmanı

: Yrd. Doç. Dr. A. Gökhan YAVUZ

Prof. Dr. Bülent ÖREN

Prof. Dr. Dya Kalpsiz

İSTANBUL, 2002

128644

İÇİNDEKİLER

Sayfa

SİMGE LİSTESİ	vi
KISALTMA LİSTESİ	vii
ŞEKİL LİSTESİ	viii
ÇİZELGE LİSTESİ	x
ÖNSÖZ	xi
ÖZET	xii
ABSTRACT	xiii
1. GİRİŞ	1
2. DOKÜMAN YÖNETİM SİSTEMLERİ	3
2.1 Fiziksel Doküman Yönetimleri İle İlgili Problemler	4
2.2 Doküman Yönetim Sisteminin Getirdikleri	5
2.3 Elektronik Döküman Yönetim Sistemlerine Neden İhtiyaç Duyarız?	5
2.4 Dokümanın Yönetim Sistemlerinin Önemi	7
2.5 Kağıt ve Dijital Dökümanları Bir Sistem Altında Toplamak	7
2.5.1 Döküman nedir?	8
2.5.2 Dijital döküman nedir?	9
2.6 Bankacılıkta Entegre Doküman Yönetimi Sistemleri	9
2.6.1 Entegre doküman yönetim sistemleri ile gelen büyük değişim	11
2.6.2 Entegre doküman yönetim sistemi seçimi	11
2.6.2.1 Güvenilirlik ve güvenlik	12
2.6.2.2 Felaket Senaryosu	12
2.6.2.3 Performans	12
2.6.2.4 Mainframe entegrasyonu	12
2.6.2.5 Uygulama entegrasyonu	13
2.6.2.6 İşakışı (workflow) entegrasyonu	13
2.7 Sabit Disk Maliyetleri	14
2.8 Doküman Yönetim Sistemi Genel Kavramları	15
2.8.1 Anahtar alanları	16
2.8.2 İndeksleme	16
2.8.3 İndeksleme sınıfları	16
2.8.4 Klasörler	16
2.8.5 Çalışma listeleri	16
2.8.6 Giriş kontrol listeleri	16
2.8.7 Ayrıcalıklar	16
3. TASARIM	18
3.1 Sistemde Olması Gereken Modüller	18

3.2	Tasarlanan Sistem Yapısı	21
3.3	Sistem Mimarisi.....	22
3.4	Sistemin Temel Yapısının Anlatılması.....	24
3.5	Ne Başarmak İstiyoruz?.....	25
4.	KÜTÜPHANE SUNUCU MİMARİSİ	29
4.1	Kütüphane Veritabanı Tablolarının Oluşturulması	30
4.2	Kütüphane Sunucusunun Veritabanı Neler Olabilir?	31
4.3	Veritabanı Sistem Bilgisinin Oluşturulması	32
4.3.1	Veritabanı kolon tipleri.....	32
4.3.2	Giriş kontrol listeleri.....	33
4.3.3	Giriş kontrol liste detayları	33
4.3.4	Anahtar alanları	34
4.3.5	Güncellemenin senkron hale getirilmesi	35
4.3.6	İndeks sınıflarının anahtar alanları	36
4.3.7	İndeks sınıfları	36
4.3.8	Doküman tiplerinin belirlenmesi	37
4.3.9	Sistem kütüğü	38
4.3.10	Nesne not bilgileri.....	39
4.3.11	Dokümanların fiziksel yer bilgilerinin tutulması.....	39
4.3.12	Sistem katalog bilgisi.....	40
4.3.13	Klasör doküman hiyerarşisi	40
4.3.14	Sistem parametrelerinin tanımlanması	41
4.3.15	Doküman sunucu disk parçaları bilgisi.....	43
4.3.16	Doküman sunucu disk bilgisi.....	44
4.3.17	Doküman sunucu bilgileri	44
4.3.18	Kullanıcı grupları.....	45
4.3.19	Sistem Kullanıcıları	45
4.3.20	Ayrıcalık setleri	47
4.3.21	SMS sunucu seçeneklerinin belirlenmesi	48
4.3.22	Çalışma liste detayları.....	50
4.3.23	Çalışma listeleri	51
4.4	Veritabanı Dinamik Tabloları.....	52
4.5	Statik Prosedürler.....	53
4.5.1	Yönetici program kütüphane sunucusu statik prosedürleri	53
4.5.2	SMS Sunucu kütüphane sunucusu statik prosedürleri.....	57
4.5.3	İstemci kütüphane sunucusu statik prosedürleri	58
4.6	Dinamik Olarak Yaratılan Statik Prosedürler.....	62
4.6.1	NOINDEX	63
5.	DÖKÜMAN SUNUCULARIN MİMARİSİ.....	64
5.1	Doküman Sunucu Üzerindeki FTP Sunucu.....	64
5.2	Doküman Sunucu Disklerinin Ulaşımına Açılması.....	64
5.3	Doküman Alt Parçaları	67
5.4	Mantıksal Disk Parçalarının Oluşturulması ve Kullanılması	68
5.5	Versiyonlama	72
5.6	Yedekleme	72
5.7	SMS Sunucu	73
6.	STANDART İSTEMCİ.....	76

6.1	Araç Çubukları.....	77
6.2	Çalışma Listelerinin Listelenmesi	79
6.3	Arama	80
6.4	Klasör Yaratılması	83
6.5	Klasörlerin Listelenmesi	84
6.6	İmaj Dosyaların Gösterilmesi	85
6.7	Sisteme Doküman Eklenmesi	86
6.8	Farklı Tipte Dosyayı Uygulama İçinde Açma.....	87
6.9	Farklı Tipte Dosyayı Uygulama Dışı Bir Programla Açma	88
7.	YÖNETİCİ PROGRAM.....	90
7.1	Yönetici Program Nedir?	90
7.2	Anahtar Alanlarının Tanımlanması	90
7.3	Ayrıcalık Setleri.....	91
7.4	Giriş Kontrol Listeleri.....	91
7.5	Kullanıcıların Tanımlanması	94
7.6	Grupların Yönetilmesi	94
7.7	İndeks Sınıfları.....	95
7.8	Çalışma Listeleri	95
7.9	Sistem Konfigürasyonu	95
7.10	Doküman Sunucuları ve Disk Parçacıkları.....	96
7.11	SMS Seçenekleri.....	96
7.12	Sisteme Giriş.....	96
7.13	Anahtar Alanlar	97
7.14	Ayrıcalıklar	97
7.15	Giriş Kontrol Listeleri.....	98
7.16	Sistem Kullanıcıları	99
7.17	Gruplar	100
7.18	İndeks Sınıfları.....	101
7.19	Çalışma Listeleri	102
7.20	Sistem Konfigürasyonu	103
7.21	Doküman Sunucuların Belirlenmesi ve Disk Parçalarının Oluşturulması	104
7.22	SMS Seçeneklerinin Belirlenmesi	105
8.	PERFORMANS.....	107
8.1	Çalışma Listelerinin Listelenmesi	107
8.2	Klasörlerin Listelenmesi	110
8.3	Arama Performansı	111
8.4	Sistem Performans Testleri.....	111
8.5	Sisteme Doküman Ekleme Performansı	112
8.6	Genel Sistem Performansı	112
9.	SONUÇ.....	114
	KAYNAKLAR.....	117
	EKLER	118
	Ek 1 Neler geliştirilebilir?	119
	Ek 2 Akış Diagramları	120



SİMGE LİSTESİ

χ	Set elemanlarından biri
$\vee$	Veya
$+$	Elemanlarından oluşturulan bir veritabanı kaydı
ψ	Veritabanı kayıtlarından oluşturulan veri seti



KISALTMA LİSTESİ

ÇL	Çalışma Listesi
DMS	Document Management System
DS	Doküman Sunucusu
DY	Doküman Yönetim
DYS	Doküman Yönetim Sistemi
KS	Kütüphane Sunucusu
SMS	Yer Yönetici Sunucusu
SP	Statik Prosedür
VTYS	Veri Tabanı Yönetim Sistemi
YP	Yönetici Program



ŞEKİL LİSTESİ

sayfa

Şekil 2.1	Dokümanın firma içi süreci	6
Şekil 3.1	Sistemin basit modül şeması.....	22
Şekil 3.2	Sistemin detaylı modül yapısı.....	23
Şekil 3.3	Sistemin çalışma şekli.....	25
Şekil 3.4	Dokümanların standart istemci uygulamasında indekslenmesi	26
Şekil 3.5	Dokümanların standart istemci uygulamasında gösterilmesi	27
Şekil 3.6	Klasörlerin standart istemci uygulamasında indekslenmesi	27
Şekil 3.7	Klasörleri standart istemci uygulamasında listelenmesi	28
Şekil 4.1	Kütüphane sunucusu mimarisi.....	29
Şekil 5.1	Takma ad ile farklı disklerin gösterilmesi	65
Şekil 5.2	İşletim sistemi seviyesinde klasörün disklere linklenmesi gösterilmesi.....	65
Şekil 5.3	Doküman alt parçalarının oluşturulması.....	67
Şekil 5.4	Doküman sunucu disk parçaları.....	69
Şekil 6.1	Araç çubukları - 1	77
Şekil 6.2	Araç çubukları - 2.....	78
Şekil 6.3	Çalışma listeleri ve detayları.....	80
Şekil 6.4	Tüm indekslerde arama yapma	81
Şekil 6.5	Tüm indekslerde arama sonuçları	82
Şekil 6.6	Tek bir indeks sınıfında arama.....	83
Şekil 6.7	Arama sonuçları	83
Şekil 6.8	Klasör yaratma	83
Şekil 6.9	Klasör bulma.....	84
Şekil 6.10	Klasör listeleme	85
Şekil 6.11	Dokümanların gösterilmesi	86
Şekil 6.12	Dokümanların güncellenmesi	86
Şekil 6.13	Sisteme doküman ekleme	87
Şekil 6.14	Bir excel dokümanın açılması.....	88
Şekil 6.15	Bir MP3 dosyasının açılması	89
Şekil 7.1	Sisteme giriş ekranı.....	96
Şekil 7.2	Anahtar alanların tanımlanması	97
Şekil 7.3	Ayrıcalık setlerinin tanımlanması	98
Şekil 7.4	Giriş kontrol listelerinin tanımlanması	99
Şekil 7.5	Sistem kullanıcılarının tanımlanması.....	100
Şekil 7.6	Grupların tanımlanması	101
Şekil 7.7	İndeks sınıflarının tanımlanması.....	102
Şekil 7.8	Çalışma listelerinin tanımlanması.....	103
Şekil 7.9	Sistem konfigürasyon paneli.....	104
Şekil 7.10	Doküman sunucularının tanımlanması.....	105
Şekil 7.11	SMS seçeneklerinin belirtilmesi	106
Şekil 8.1	Çalışma listesi yenileme	108
Şekil Ek.2-1	Createdocument metodu akış diagramı.....	120
Şekil Ek.2-2	Createfolder metodu akış diagramı.....	121
Şekil Ek.2-3	Logon metodu akış diagramı - 1	122
Şekil Ek.2-4	Logon metodu akış diagramı - 2	123
Şekil Ek.2-5	Çalışma listeleri prosedürü akış diagramı.....	124
Şekil Ek.2-6	İndeks sınıfları prosedürü akış diagramı.....	125
Şekil Ek.2-7	Search metodu akış diagramı - 1.....	126

Şekil Ek.2-8	Search metodu akış diagramı - 2.....	127
Şekil Ek.2-9	Addpart metodu akış diagramı.....	128
Şekil Ek.2-10	Addtofolder metodu akış diagramı	129
Şekil Ek.2-11	Checkin metodu akış diagramı.....	130
Şekil Ek.2-12	Checkout metodu akış diagramı.....	131
Şekil Ek.2-13	Createpart metodu akış diagramı	132
Şekil Ek.2-14	Delete metodu akış diagramı	133
Şekil Ek.2-15	Deletepart metodu akış diagramı	134
Şekil Ek.2-16	Deletepartversion metodu akış diagramı	135
Şekil Ek.2-17	Gethistorylog metodu akış diagramı.....	136
Şekil Ek.2-18	Getnotes metodu akış diagramı.....	137
Şekil Ek.2-19	Getparentfolders metodu akış diagramı.....	138
Şekil Ek.2-20	Getpart metodu akış diagramı.....	139
Şekil Ek.2-21	Nextworkbasketitem metodu akış diagramı.....	140
Şekil Ek.2-22	Open metodu akış diagramı	141
Şekil Ek.2-23	RefreshTOC metodu akış diagramı	142
Şekil Ek.2-24	Removefromfolder metodu akış diagramı	143
Şekil Ek.2-25	Removefromwb metodu akış diagramı	144
Şekil Ek.2-26	Routetowb metodu akış diagramı	145
Şekil Ek.2-27	Saveindex metodu akış diagramı – 1	146
Şekil Ek.2-28	Saveindex metodu akış diagramı - 2.....	147
Şekil Ek.2-29	Saveindex metodu akış diagramı – 3	148
Şekil Ek.2-30	Savenotes metodu akış diagramı.....	149
Şekil Ek.2-31	Setpriority metodu akış diagramı.....	150

ÇİZELGE LİSTESİ

sayfa

Çizelge 2.1	1992-2010 yılları arası sabit disk maliyet tahmini.....	14
Çizelge 4.1	NOINDEX tablo yapısı.....	31
Çizelge 4.2	DMSACCESSLISTCODES tablo yapısı.....	33
Çizelge 4.3	DMSACCESSLISTS tablo yapısı	33
Çizelge 4.4	DMSATTRIBUTEDEFINITIONS tablo yapısı	34
Çizelge 4.5	DMSCHECKEDOUT tablo yapısı	35
Çizelge 4.6	DMSCLASSATTRIBUTES tablo yapısı	36
Çizelge 4.7	DMSCLASSDEFINITIONS tablo yapısı	36
Çizelge 4.8	DMSCONTENTCLASSES tablo yapısı	37
Çizelge 4.9	DMSHISTORYLOG tablo yapısı.....	38
Çizelge 4.10	DMSITEMNOTE tablo yapısı.....	39
Çizelge 4.11	DMSITEMOBJECTS tablo yapısı.....	39
Çizelge 4.12	DMSITEMS tablo yapısı	40
Çizelge 4.13	DMSLINKS tablo yapısı.....	41
Çizelge 4.14	DMSMASTER tablo yapısı	41
Çizelge 4.15	DMSOBJECTSERVERPARTITIONS tablo yapısı	43
Çizelge 4.16	DMSOBJECTSERVERDISCS tablo yapısı	44
Çizelge 4.17	DMSOBJECTSERVERS tablo yapısı	44
Çizelge 4.18	DMSPATRONGROUP tablo yapısı.....	45
Çizelge 4.19	DMSPATRONS tablo yapısı	45
Çizelge 4.20	DMSPRIVILEGES tablo yapısı	47
Çizelge 4.21	Ayrıcalık Detay Açıklamaları	47
Çizelge 4.22	DMSSMSOPTIONS tablo yapısı	48
Çizelge 4.23	DMSWIPITEMS tablo yapısı	50
Çizelge 4.24	DMSWORKBASKETS tablo yapısı	51
Çizelge 4.25	Örnek dinamik tablo yapısı.....	53
Çizelge 4.26	YP tarafından kullanılan statik prosedürleri	53
Çizelge 4.27	SMS tarafından kullanılan statik prosedürleri	57
Çizelge 4.28	İstemci tarafından kullanılan statik prosedürler	58
Çizelge 5.1	Örnek item nesne tablosu (DMSITEMOBJECTS).....	67
Çizelge 5.2	Örnek DMSOBJECTSERVERS tablosu	69
Çizelge 5.3	Örnek DMSOBJECTSERVERDISCS tablosu.....	69
Çizelge 5.4	Örnek DMSOBJECTSERVERPARTITIONS tablosu	70
Çizelge 5.5	Örnek DMSITEMOBJECTS tablosu.....	71
Çizelge 5.6	SMS sunucu seçenek tablosu	73
Çizelge 8.1	Performans makina konfigürasyonu	111
Çizelge 8.2	Sistem doküman yaratma performansı.....	112
Çizelge 8.3	Önemli fonksiyonların ortalama gerçekleştirilme süreleri	113

ÖNSÖZ

İki yıldan fazla bir süredir Türkiye'nin bilişim sektöründe öncü firmalarından Garanti Teknoloji'nin Doküman Yönetimi ve İş Akışı biriminde çalışıyorum. Garanti Teknoloji doküman yönetim projesine ilk başladığında Türkiye'de istenen fonksiyonları tümüyle karşılayacak bir doküman yönetim sistemi mevcut değildi.

Fakat iki seneden bu yana teknoloji her zamanki gibi çok hızlı gelişti ve doküman yönetim sistemleri firmalar için vazgeçilmez hale geldiler. Çünkü firmalar kağıda dayalı arka ofis operasyonları için bütçelerinin %30-%50'sini ayırmak zorundadırlar. Buna ek olarak fiziki doküman ile çalışmak üzere kurulu doküman yönetim sistemlerinde birtakım dezavantajlar da vardır. Örneğin, dokümanlara istenildiği anda ulaşmak kolay değildir. Bir sürecin tamamlanması için elden ele dolaşan dokümanlara kontrollü bir akış mantığı verilememekte, planlama yapılamamakta, dokümanların güvenliği sağlanamamakta ve gereğinden fazla personel kullanmak durumunda kalınmaktadır. Doküman yönetimi karmaşık hale geldikçe hata olasılığı ve hatalardan kaynaklanan müşteri şikayetleri de bir o kadar artmaktadır.

Diğer yandan doküman yönetim sistemine sahip olmayan firmaların karşılarında kağıt masrafı sorunu vardır. Tabii ki dokümanların arşivlenme sorunu da firmaların çözmesi gereken diğer bir problem olarak karşımıza çıkmaktadır. Milyonlarca doküman hem çok geniş bir yerin kiralınmasını hem de istenildiğinde kolay erişim için gerekli düzenlemenin yapılmasını gerektirmektedir.

Bilgisayar çağında ise tabii ki bunlara elektronik ortamda bir çözüm bulmak mümkündür. Bu çözümü getiren uygulamalar doküman yönetim sistemleri olarak adlandırılır. Şurası şüphe götürmez bir gerçek ki, dokümanların elektronik ortamda görüntülenmesi, yönlendirilmesi ve yönetilmesi firmalar için çok büyük bir avantajdır. Firmaların müşterilerine sunduğu kalite farkını kendi iş akışlarının ve sahip oldukları bilgisayar sistemlerinin hızlarının belirlediği bu şiddetli rekabet ortamında, doküman yönetim sistemlerinden uzak kalmaları, firmanın ticari geleceğini etkileyecek bir eksikliktir.

ÖZET

Günümüzde varolan şirketlerin çoğunun yıllık harcamalarının %30 ile %50'si kağıda dayalı günlük geri plan ofis operasyon masrafları için kullanılmaktadır. Yürütülen bu operasyonun en büyük parçasını kağıt üzerinde tutma masrafları ve dokümanların düzenli bir şekilde arşivlenmesi oluşturur.

Doküman yönetim sistemlerinin başlıca amacı şirketleri kağıt masrafından kurtarmak, dokümanları elektronik ortamda tutarak bir düzen getirmek ve daha verimli bir şekilde kullanıma açmaktır.

Doküman yönetim sistemleri bir çok doküman tipi için kullanılabilir. Ancak başlıca kullanım amacı faks ve tarayıcıdan geçirilmiş imaj dosyalarıdır.

Doküman yönetim sistemleri bununla beraber tüm dokümanlar için bir depo görevi görebilir. PDF dosyalardan DOC word dosyalarına, Excel tipli dokümanlardan PPT uzantılı sunumlara kadar herşeyi içinde belirli bir mantıkla depolayabilir, verimli bir şekilde kullanıma açabilir.

Vizyonu genişletecek olursak büyük bir kütüphaneyi PDF dosyalar şeklinde doküman yönetim sisteminde indekslemek mümkündür.

Bir başka örnek vermek gerekirse, sistemi kullanan bir kullanıcı kendi görüntüsünü kaydettiği bir MPEG görüntü dosyasını bir başka arkadaşının iş listesine yollayabilir ve arkadaşı da bu MPEG dosyayı açarak izleyebilir. Tabii ki bu dosyanın WAV ses veya herhangi bir tipte dosya olması da mümkündür.

Şu kaçınılmaz bir gerçektir ki, çağımız şirketleri bir doküman yönetim sistemine gereksinim duyarlar. Bu tezde tasarlanan sistem bu doküman yönetim sistemlerinden bir tanesidir.

ABSTRACT

Today's firms spend %30-%50 of their annual budget on the operation of their daily works. The most expensive part of the operation is the money spent on the Document Management and Workflow.

The main reason why Document Management System is definately needed by firms is to get rid of spending that much money on paper by having documents on electronic environment. So firms can easily handle the workflow of their documents and make the best use of them.

Document Management Systems can be used for many document types. But the main area of DMS usage is to store faxes and scanned images.

Document Management Systems can also be a storage for all document types that the operating systems can handle on which DMS is running. From PDF files to DOC word files, from Excel file types to PPT presentation powerpoint documents, Document Management Systems can efficiently store every document types.

If we enlarge the vision, a library with thousands of books can be indexed in Document Management Systems.

As an another example, a user in the system can record his video and put this record into his friend's work list. His friend opens the video coming from his colleague and watches the video stream. Of course, this file can be a WAV file which records voices.

All firms definitely need a Document Management System. The system designed in the thesis is one of these Document Management Systems.

1. GİRİŞ

Hazırlanan bu tez, bir doküman yönetim sistemi tasarımı ve uygulamasıdır. Doküman Yönetim Sistemleri (DYS), fiziksel dokümanların elektronik ortama alınmasını, elektronik ortamda paylaşımını, şirket içinde dolaşımını ve yönetimini sağlar.

Sağlıklı bir bilgi yönetimi, kurumların ihtiyaçları doğrultusunda belirlenecek yöntemle, oluşturdukları dokümanları doğru bir şekilde yönetebilmesi üzerinde şekillenmektedir. Kağıt üzerindeki şirket dokümanlarının paylaşımı, şirket içinde dolaşımı ve yönetimi fazla işgücü kaybına sebep olan ve zaman zaman yapılması gereken gerçek "iş" ten daha fazla zaman alan bir süreçtir.

Şirketlerin günlük iş süreçlerinde ortaya çıkan bilginin büyük bir çoğunluğu doküman formundadır. Bunlar; mektuplar, faksler, sözleşmeler, dosya notları, faturalar, irsaliyeler, e_postalar, internetten indirilen HTML sayfaları, taranan imaj dosyaları, fotoğraflar, ses ve görüntü dosyaları, sunum dosyaları, word ve excel dosyaları gibi birbirinden farklı şekillerde oluşturulan doküman türleridir. Bilgisayarların iş süreçlerine dahil edilmesiyle, doküman trafiği azalmamış; aksine daha da artmıştır. İnternetin de devreye girmesiyle bugün her kuruluş, karmaşık bir doküman trafiği içinde sıkışmış; ve sayıları her geçen gün çoğalan dokümanlarını yönetebilmekte büyük sorunlar yaşamaya başlamıştır.

Fatura veya irsaliye verileri gibi dokümanlar muhasebe süreci içinde yönetildiği için, sözkonusu doküman trafiğinin biraz dışında kalsalar da; bunun dışında yer alan ve şirketin iş süreçlerine önemli bir temel teşkil eden diğer dokümanlar sistemli bir şekilde yönetilememektedir. Bu aşamada, doküman yönetimi, şirketlerin verimini artırabilecek büyük bir potansiyeli barındırmasına rağmen, çoğu zaman gözden kaçmaktadır. Ancak unutulmaması gereken nokta; doküman yönetiminin her zaman, bilginin sağlıklı bir şekilde yönetilebilmesinin en temel unsuru olduğudur.

Doküman yönetiminin önemli bir ekonomik getiriye sahip olduğu, IDC'nin Avrupa ve Amerika'daki şirketlerde yaptığı araştırmalarda ispatlanmıştır. Dokümanların merkezi arşivlerde saklanması ile bilginin lokalize edilmesi önlenirken; bu dokümanlara erişim için firmanın başka bir yerine gidilmesi zorunluluğu da ortadan kalkmıştır. Böylece, dokümanların saniyelerle ölçülebilecek sürelerde, mekanlardan bağımsız olarak kişilerin kullanımına sunulması ile herhangi bir mekana giderek onlara ulaşılması arasındaki verim artışında çok büyük farklılıklar oluşmuştur. Arama süresinin uzaması, daha verimli çalışmalarda kullanılabilecek iş gücü kaybı demektir. Farklı araştırmaların sonuçlarına göre; bir çalışan

vaktinin yarısından çoğunu, dokümanları aramak ve onları aldığı yere geri koymakla geçirmektedir.

Doküman Yönetim Sistemleri, kağıt ortamında veya diğer optik araçlar üzerinde doküman arşivleme çözümleriyle karşılaştırıldığında birçok avantaj sunmaktadır. Çünkü doküman yönetim sistemleri, sadece bir arşivleme yöntemi değil; aynı zamanda bir araştırma ve iş akış sistemi olarak da farklı artı değerler getirmektedirler. Bu artı değerlerden en önemlisi, dokümanların düzeninin bozulmaması ve onlara sürekli ulaşabilmektir. Dokümanlar aynı anda birden fazla kişinin kullanımına açılır. Dokümanlara çok daha hızlı ulaşabilmek mümkün hale gelir. Rahat arama olanakları sağlanır. Fiziksel olarak daha az bir yere ihtiyaç vardır. Farklı doküman tipleri kaydedilip saklanabilir. Bilişim teknikleriyle uyum üst seviyededir. Getirdiği zaman ve mekan avantajları şirket için hemen artı değere dönüşür.

Doküman yönetim sistemi bir dokümana; oluşum, değişiklik yapma ve arşivlenme süreçlerinin tamamında eşlik etmeyi öngörür. Doküman, yaşam sürecinin her anında sistemin güvenlik mekanizmalarının kontrolünde kalmaktadır. Dokümanlar karşımıza kağıt üzerinde veya dijital olarak çıkmaktadır.

Dijital ortamlardaki dokümanlar, bilgisayarlarda yazı programları veya hücresel hesaplama programlarıyla üretilmektedir. Dışarıdan gelen faksler veya taranan dokümanlarda sistemde olabilecek dijital doküman tiplerindendir. Doküman yönetim sistemleri, klasik arşivleme sistemlerinin aksine tam bu noktada önemli bir açığı kapatmaktadır. Bir doküman oluşturulduğu gibi sistem tarafından, birer okunabilir CI-Doküman (Coded Information) olarak tanımlanır ve kaydedilir. Bu işlem otomatik olarak gerçekleşmektedir; çünkü sistem, standart ofis uygulamalarıyla bütünleşik çalışmaktadır. Bu entegrasyon sadece dokümanların kayıt aşamalarında değil, arama işlemlerinde de direkt olarak devreye girmektedir. Bunun yanında, doküman yönetim sistemiyle bütünleşik herhangi bir uygulamada yaratılmış dokümanlardan başka, taranmış dokümanları da yönetmek mümkündür.

Doküman yönetim sistemi, verimli bir iş akışı oluşturabilmek için ön koşuldur. Dokümanların, çalışan grupları tarafından ortak kullanıma açılması ve bunların elektronik ortamda paylaşımları, şirketlerin süreç kalitelerinin önemli oranda iyileşmesini sağlamaktadır. Burada özellikle yönetim yöntemi üzerinde durulması, doküman yönetim sisteminin bir ürün değil, şirket organizasyonlarında mevcut kurallar ve programlar bütünü olmasından kaynaklanmaktadır.

2. DOKÜMAN YÖNETİM SİSTEMLERİ

Doküman Yönetim Sistemleri, fiziksel dokümanların elektronik ortama alınmasını, elektronik ortamda paylaşımını, şirket içinde dolaşımını ve yönetimini sağlayan sistemlere verilen addır.

Sağlıklı bir bilgi yönetimi, kurumların ihtiyaçları doğrultusunda belirlenecek yöntemle, oluşturdukları dokümanları doğru bir şekilde yönetebilmesi üzerinde şekillenmektedir. Kağıt üzerindeki şirket dokümanlarının paylaşımı, şirket içinde dolaşımı ve yönetimi fazla işgücü kaybına sebep olan ve zaman zaman yapılması gereken gerçek "iş" ten daha fazla zaman alan bir süreçtir. (http://www.enter.com.tr/web/eiscozumleri/index_dokuman_yonetimi.asp?secim=eiscozumleri04).

Günümüzde her kuruluş, karmaşık bir doküman trafiği içinde sıkışmış; ve sayıları her geçen gün çoğalan dokümanlarını yönetebilmekte büyük sorunlar yaşamaya başlamıştır.

Doküman yönetiminin önemli bir ekonomik getiriye sahip olduğu, IDC'nin Avrupa ve Amerika'daki şirketlerde yaptığı araştırmalarda ispatlanmıştır. Dokümanların merkezi arşivlerde saklanması ile bilginin lokalize edilmesi önlenirken; bu dokümanlara erişim için firmanın başka bir yerine gidilmesi zorunluluğu da ortadan kalkmıştır. Böylece, dokümanların saniyelerle ölçülebilecek sürelerde, mekanlardan bağımsız olarak kişilerin kullanımına sunulması ile herhangi bir mekana giderek onlara ulaşılması arasındaki verim artışında çok büyük farklılıklar oluşmuştur.

Firmaların fiziksel arşiv sorunlarını çözmek ve elektronik ortamda doküman yönetimlerini gerçekleştirmek ve geliştirmek için ortaya konulan bu modele elektronik doküman yönetim sistemi adı verilir.

Döküman Yönetimi genel olarak iki başlık altında toplanabilir.

1. Fiziksel Döküman Yönetimi
2. Elektronik Döküman Yönetimi

Fiziksel döküman yönetimi kağıt üzerindeki şirket dokümanlarının paylaşımı, şirket içinde dolaşımı ve yönetimidir. Fiziksel döküman yönetim sisteminde kağıt üzerindeki şirket dokümanlarının paylaşımı, şirket içinde dolaşımı ve yönetimi fazla işgücü kaybına sebep olan ve zaman zaman yapılması gereken gerçek "iş" ten daha fazla zaman alan bir süreç olduğu için önemli dezavantajları olan bir döküman yönetim sistemidir.

Elektronik döküman yönetim sistemi ise fiziksel dokümanların elektronik ortama alınmasını, elektronik ortamda paylaşımını, şirket içinde dolaşımını ve yönetimini sağlar.

Bu bölümde Döküman Yönetim Sistemleri için tüm dünyadaki bilimsel ve ticari genel bakış anlatılacak ve fiziksel veya elektronik döküman yönetim sistemleri üzerine yazılmış bazı makalelerden söz edilecektir.

2.1 Fiziksel Doküman Yönetimleri İle İlgili Problemler

Günümüz şirketlerinin kağıt üzerindeki şirket dokümanlarının paylaşımı, şirket içinde dolaşımı ve yönetimi fazla işgücü kaybına sebep olan ve zaman zaman yapılması gereken gerçek "iş" ten daha fazla zaman alan bir süreçtir. Bundan dolayı, şirket içi fiziksel doküman yönetimi ile ilgili bir çok problem mevcuttur.

Bu problemlerden en önemlisi fiziki dokümana aynı anda sadece bir kullanıcının ulaşabilmesidir. Bu, işyeri için bir dezavantajdır, çünkü hem iş hem de zaman kaybına sebep olur ve iş verimli bir şekilde gerçekleştirilemez.

Bir diğer problem dokümanların masadan masaya taşınmasında kaybolma riskidir. Aynı zamanda güvenlik yok denecek kadar azdır. Doküman ortada gezerken görmemesi gereken kişiler dokümana rahatlıkla ulaşabilirler.

Önemli problemlerden bir diğeri aranılan doküman veya klasöre hızla ve kolayca ulaşabilme problemidir. Özellikle bu iş, doküman veya klasör arşivdeyse bir kat daha zorlaşır çünkü çok fazla doküman arasından bulmak zor hale gelir. Bu zaman azımsanmayacak kadar fazladır.

Şirket içi en büyük zaman kayıpları dokümanların klasörlenmesinde yaşanır. Orjinal doküman sadece bir klasörde yer alabilir, yada kopyalanarak birden fazla klasöre konacaktır. Bu da hem zaman kaybına, hem de kağıt masrafına neden olur.

Dökümanlar veya klasörler bir şirket içi bir iş akışına kolayca sokulamazlar. Prosesin yürütmesinde bir dokümanın veya klasörün bir kişiden başka bir kişiye ulaşma gerensinimi vardır. Ancak bu iş, özellikle doküman veya klasörler fiziksel olarak tutulduklarında, elektronik ortamdakinden çok daha fazla zaman alır. Bu problemlerden dolayı firma üretimi doğru şekilde ölçemez. Dokümanların ne tür işlemlere maruz kaldığının ve kimler tarafından değiştirildiğinin tarihçesini tutmak çok zordur.

Ayrıca kağıt mafrası inanılmaz boyutlardadır. Bu masraf firmanın tipine göre yıllık bütçenin %20'si ile %30'u arasında olabilir.

Klasörler için bir indeksleme yoktur. Oysa doküman yönetim sistemlerinde klasörlerde dokümanlar gibi indekslenirler. Bu yöntem onları bulmayı çok kolaylaştırır.

Bir yangın sırasında tüm arşivi kaybetmek mümkündür. Tüm doküman ve klasörlerin fiziksel olarak kopyalarını sürekli yedeklemek ise mümkün olmayan bir iştir.

2.2 Doküman Yönetim Sisteminin Getirdikleri

Doküman yönetim sistemleri, sadece bir arşivleme yöntemi değil; aynı zamanda bir araştırma ve iş akış sistemi olarak da farklı artı değerler getirmektedirler.

Bunları aşağıdaki gibi maddeleyebiliriz.

- Dokümanların düzeninin bozulmaması ve onlara sürekli ulaşabilmek.
- Tüm şirketin dokümanlarının merkezi bir yerde durması.
- Dokümanların aynı anda birden fazla kişinin kullanımına açılması.
- Doküman başına yönetim maliyetinin azaltılması.
- Dokümanlara çok daha hızlı ulaşabilmenin mümkün hale gelmesi.
- Rahat arama olanakları sağlanması.
- Fiziksel olarak daha az bir yere ihtiyaç duyulması.
- Farklı doküman tiplerinin kaydedilip saklanabilmesi.
- Bilişim teknikleriyle uyumun üst seviyede olması.
- Getirdiği zaman ve mekan avantajlarının şirket için hemen artı değere dönüşmesi.
- Sistemdeki klasörlerin de dokümanlar gibi indekslenebilmesi.
- Sistemdeki bulunan ve kullanıcılar tarafından erişilebilen tüm kaynaklar için güvenlik maksimum seviyede sağlanması.
- Doküman veya klasör kayıplarının ortadan kaldırılması.
- Doküman ve klasörlerin tutulduğu mekanlardan %40-%70 oranında yerden kazanılması.
- Ayrıca firma içinde ortalarda dolaşan doküman sayısının %80 azalması.
- Sistem içinde sadece metin dokümanların değil tüm doküman tiplerinin yönetilebilmesi. Bu doküman tipleri arasında bir görüntü dosyasının bile olabilmesi.
- Dokümanlar için versiyonlama yapılabilmesi.
- Sistemde sürekli bir yedekleme mekanizmasının mevcut olması.
- Üretimin ölçülebilir hale getirilebilmesi.
- Kilometrelerce uzağa dokümanın anında taşınabilmesi,
- Doküman ve klasör iş akışının sistem tarafından gerçekleştirilebilmesi.

2.3 Elektronik Döküman Yönetim Sistemlerine Neden İhtiyaç Duyarız?

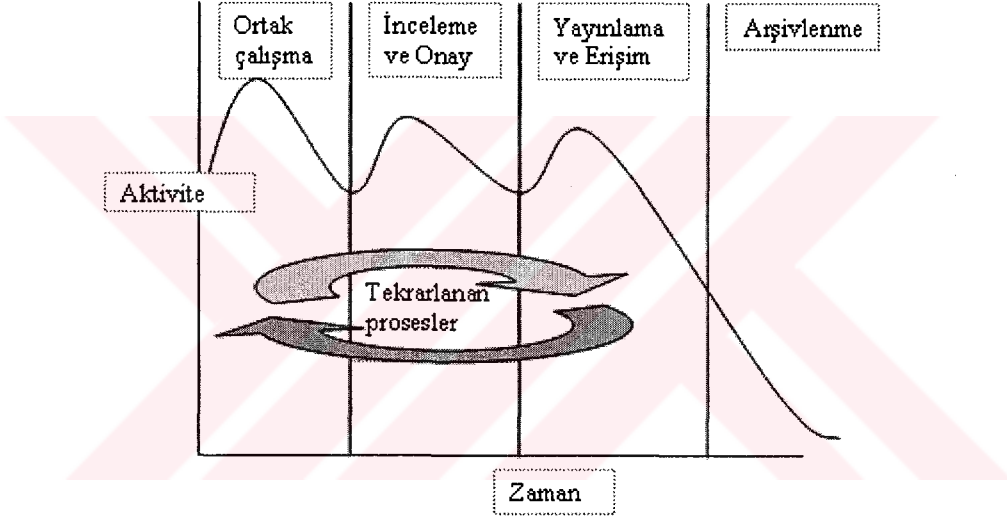
Gartner Grubu'nun 1997 yılında yaptığı araştırmaya göre, şirket çalışanları haftada 8 saatlerini (toplam çalışma süresinin %20'si) belge yönetimi için harcıyorlar. Bugün ise bu oranın çalışma saatlerinin %20-%30'unu oluşturduğu, 2003 yılı itibarıyla da %30-%40'ını alacağı tahmin ediliyor.

Bugün şirketlerde bir doküman yönetim sistemi kurmak çok ciddi maliyet azalmasını da beraberinde getiriyor. Ortalama bir çalışanın yılda 12.000\$'lık bir maaş aldığını varsayarsak

zamanının %20'si doküman yönetimi işleri ile geçtiği durumda, yılda 2.400\$ veya 100 kişilik bir şirkette 240.000\$ boşa giden harcamalar olarak görülebilir.

İnsan kaynağı maliyetlerine ek olarak elektronik veya kağıt ortamında belge işleme maliyetlerini de düşünürsek; Gartner'a göre, doküman yönetimi yapılmayan bir sistemde aşağıdaki maliyetler de söz konusu olmaktadır.

- Ortalama olarak bir belge kağıt veya elektronik ortamda 9 ile 11 kez arasında kopyalanıyor ve maliyeti 18\$.
- Belgelerin dosyalanma maliyeti belge başına 20\$.
- Yanlış dosyalanmış bir belgenin bulunmasının maliyeti 120\$.



Şekil 2.1 Dokümanın firma içi süreci

(<http://www.kets.com/tr/urunler/dokuman.yonetimi/>)

Yönetilemeyen belgeler saklı veya açık pek çok maliyet içeriyorlar ve doküman yönetim sistemleri bu maliyetleri yok etmek ya da azaltmak için çok ciddi avantajları beraberinde getiriyor.

Bir belgenin yaratılmasından, yok olmasına kadar geçen süreçleri düşünecek olursak, belgeler genel olarak şekil 2.1'deki süreçlerden geçiyorlar. Bir belge için bu süreçler bazen bir kez, bazen de birçok kez üst üste tekrar edebiliyor.

2.4 Dokümanın Yönetim Sistemlerinin Önemi

“Daha çok firma doküman yönetim sistemlerini kendi bünyelerinde uyguladıkça, elektronik doküman yönetim sistemleri firmaların yeniden yapılandırılmaları süreci içinde merkez nokta olacaktır” (Carl Frappaolo, Executive Vice President of Delphi, Delphi White Paper, Document is the Process).

Özellikle son bir kaç yıldır, bazı büyük organizasyonlar elektronik doküman yönetim sistemlerini kendi bünyeleri içlerine adapte etmeye başladılar. Diğer bir çok firma kağıt dokümanlarla boğuşmakta iken, bir yandan da yeni doküman yönetim sistemine geçişin çarelerini arıyorlar.

Dokümanlarla çok yoğun işi olan büyük firmaların elektronik doküman yönetim sistemlerini bünyelerine adapte etmelerinin büyük yararlarının ve getirdiği kalitenin farkına diğer firmalardan daha önce vardıkları şüphesizdir.

Elektronik doküman yönetim sistemlerine alıştıkça, doküman nitelik ve nicelik olarak daha farklı özellikler almaya başlamıştır. “İleri görüşlü firmalar, kendilerini elektronik doküman yönetim sistemlerinin tüm avantajlarına kavuşturacak bu sisteme geçmek için tüm sorumlulukları üstlerine almışlar ve ortaya çıkan tüm problemlere başarıyla göğüs germişlerdir” (Carl Frappaolo). Elektronik doküman yönetim sistemlerinde dokümanlar asla kağıt parçaları halinde milyonlarca dokümanı barındıran doküman dolaplarında tutulmayacak ve vasiyetnameler gibi çok nadiren ihtiyaç duyulmayacaktır, çünkü dokümanlar artık organizasyonlar için günlük hayatın vazgeçilmez bir parçası olmuşlardır, tıpkı bir işçi gibi.

“Dokümanlar prosesi yönetebilmek için bir araç değil, prosesin ana yapısını oluşturan temel direklerdir” (Carl Frappaolo).

2.5 Kağıt ve Dijital Dökümanları Bir Sistem Altında Toplamak

Bu alt bölümde fiziksel doküman sistemleri ile elektronik döküman sistemlerinin arasındaki benzerlikleri ve farkları ortaya koyan bir makaleye yer verilmiştir. (<http://www.archivebuilders.com/whitepapers/21954p.pdf>)

Fiziksel döküman yönetim sistemlerindeki tüm problemler elektronik döküman sistemlerinde de mevcuttur. Bununla beraber, elektronik döküman sistemlerinin kendine özgü bazı diğer problemleri de vardır. Döküman yönetim sistemi işiyle uğraşanlar arasındaki bir çok kimse elektronik ortamdaki döküman kataloglama ve indeksleme işinin sadece elektronik döküman

yönetim sistemlerine ait olduğunu düşünmesine rağmen, aslında bu problem fiziksel döküman yönetim sistemlerinde de mevcuttur.

Elektronik döküman yönetim sistemlerindeki problemleri fiziksel döküman yönetim sistemlerindeki problemlerden ayırmak gerekir. Bu yüzden fiziksel döküman yönetim sistemlerinin problemlerini detaylı bir şekilde araştırmak ve ortaya koymak, elektronik döküman yönetim sistemlerinin başarı oranını bir kat daha arttıracaktır.

Elektronik döküman yönetim sistemindeki organizasyon fiziksel döküman yönetim sistemlerindeki organizasyona çok benzer. Fiziksel döküman yönetim sistemlerinde olan akıllı iş akışının elektronik döküman sisteminin yapısında yerine getirilebilmesi elektronik döküman yönetim sistemini daha başarılı bir hale getirecektir.

Fiziksel ve elektronik döküman yönetim sisteminin bir arada yönetilmesi çok pahalı bir iştir. Bu yüzden en kısa sürede, maliyetin azaltılabilmesi için bir sistemin elenmesi gerekmektedir. Bu seçeneklerde ya elektronik döküman yönetim sistemine yapılan yatırımlar kaybedilecek ya da fiziksel döküman yönetim sistemindeki fiziksel dökümanlara erişim tamamen ortadan kaldırılacaktır.

Fiziksel ve elektronik döküman yönetim sistemlerindeki yönetim birbirine çok benzer olduğundan dolayı, hem kağıt hem de dijital dökümanları yönetebilecek ortak bir sisteme ihtiyaç vardır. Elektronik döküman yönetim sisteminde kağıt dökümanların yönetilmesi tarama yoluyla dijital olarak yapılabileceğinden fiziksel döküman sistemleri olası bir seçimde elenirler.

Bir kağıt parçasını dijital ortama taşımak çok kolaydır. Böylece oluşturulan bu dijital dökümanın hem tarihçesi hem de güncelleme problemleri çok basite indirgenir.

2.5.1 Döküman nedir?

Günümüzde dökümanın bir çok tanımı yapılmaktadır. Bir döküman bilginin tanınabilir kayıdır. Her tür medya tipi kullanılabilir. Bu bir ses dosyası, bir görüntü dosyası veya basit bir metin dosyası olabilir. Bilgi veriden daha fazla şeyler ifade eder. Yani bir döküman konsept, organizasyon ve analiz içerir.

Verinin kayıtları döküman değildirler. Döküman olabilmeleri için bir konsepte, bir organizasyona ve bir analize ihtiyaç duyarlar. Veriye bir örnek olarak bir DVD'nin (Dijital Çok Yönlü Disk) hata haritası verilebilir. Bir başka örnek, bir yoldan geçen arabanın yolun bir

noktasındaki hızıdır. Bu bir veridir, döküman değil.

2.5.2 Dijital döküman nedir?

Dijital dökümanlar varolan kağıt dökümanların (fiziksel dökümanların) elektronik ortamdaki versiyonlarıdır. Tezde elektronik ortamdaki bir dökümandan bahsedildiğinde dijital döküman anlamına gelmektedir.

Dijital dökümanlar arasında doğru ve akıllı bir kontrol sağlamak, kağıt dökümanlar üzerinde doğru ve akıllı kontrol sağlamak kadar önemlidir ve bu döküman yönetim konusundaki kişilerin gözden kaçırdıkları şeylerden bir tanesidir.

Dijital dökümanları kullanabilmek için dökümanın tipini bilmemiz gerekir (Örnek PDF) ve bu dijital dökümana ait tüm bitler dökümanın elektronik ortamda saklandığı depodan tamamen ve doğru bir şekilde okunmalıdır.

2.6 Bankacılıkta Entegre Doküman Yönetimi Sistemleri

“Türkiye’de orta ölçekli bir banka, yılda yaklaşık 30 milyon sayfa kağıt doküman üretmektedir.” Kets (<http://www.kets.com/tr/urunler/dokuman.yonetimi/>)

Rekabetin ön plana çıktığı bankacılık sektöründe, hedef kitlesi hangi kesim olursa olsun, her banka, değişen müşteri ihtiyaçlarına ve beklentilerine cevap verme gerekliliğinin ve müşteri memnuniyetini sağlamanın önemini kavramıştır. Müşteriye kaliteli ve hızlı hizmet verme gereksinimi, dünyada olduğu gibi Türkiye’deki bankaları da yeni teknolojileri en etkin biçimde kullanmaya zorlamaktadır. Böylece bankalar, ister istemez yeni teknolojilerin ilk uygulayıcısı konumuna gelmektedirler.

Bankalar, geniş hizmet yelpazeleri ve müşteri tabanları dolayısıyla, ofis ortamında en fazla kağıt üreten kurumlardır. Hergün bankalara dışarıdan veya şubelerden, faks veya mektup şeklinde gelen kağıt dokümanların yanında; banka içinde üretilen raporlar, işlem makbuz ve belgeleri, yazışmalar, faturalar gibi kağıt dokümanlar, bankaların arşivlerinde, ulaşılması imkansız kağıt yığınları haline gelmektedirler. Kağıt dokümanları ayırmak, sıralamak, kopyasını almak, dosyalamak, gerektiğinde aramak, banka personeli için çok zahmetli ve zaman alıcı olmasının yanında; bu işlemler, banka için, sürekli artan işgücü, depolama ve kırtasiye giderleri anlamına gelmektedir. Dolayısıyla, bankalar gibi doküman-yoğun iş ortamlarında, yapılan işin hacmi ve maliyeti büyüdükçe, doküman yönetim sistemlerine duyulan ihtiyaç artmaktadır.

Yeni nesil Entegre Doküman Yönetim (EDY) Sistemleri, bu tür kurumlarda zamanla oluşan, birbirinden kopuk adacıklar haline gelmiş bilgi yığınlarının yönetimine ve etkin bir şekilde kullanımına olanak vermektedir. Tipik bir EDY Sistemi; **Doküman Arşiv ve Yönetim, İş-Akışı** ve **COLD** (Computer Output To Laser Disc) sistemlerinden oluşmakta; optik doküman tarayıcılar, optik saklama üniteleri, faks sistemleri gibi yazılım ve donanım birimleri tarafından tamamlanmaktadır. Bu sistemlerde; faks, kağıt belge, video, ses, HTML sayfalar, mikrofilm, EDI belgeleri, e-mail mesajları, bilgisayar/mainframe raporları, CAD çizimleri ve sayısı 250'yi aşan elektronik dosya (word, excel v.b.) tipi, genel anlamda, "doküman" olarak adlandırılmakta ve benzer şekilde işlenip yönetilmektedir.

Bankalarda kullanılan EDY sistemleri, genellikle birbirinden bağımsız çalışır, fakat entegre edilmek istendiğinde üç modülden oluşur: **Elektronik Doküman Yönetim ve Arşiv Modülü**, bankanın sahip olduğu her tür dokümanın taranarak sisteme alınmasını; indekslenmesini; işlenmesini; üzerine açıklayıcı notlar, resimler, tablolar eklenmesini; görüntülenmesini; saklanmasını; indeks bilgilerine göre aranmasını; LAN/WAN/Internet/Intranet üzerinden paylaşılmasını; faks veya e-mail ile dünyanın herhangi bir yerine gönderilmesini; iş akışı entegrasyonu sayesinde iş akışına dahil edilmesini sağlamaktadır. Bu durumda, orijinal kağıt dokümanlar ise, bankanın depolarında sadece hukuki zorunluluktan dolayı saklanmaktadır. EDY sistemlerinin **İş-Akışı** modülü ise, banka dokümanlarının belirli kurallara göre, banka içerisinde (merkez binada, merkez ile şubeleri veya müşterileri arasında) akmasına olanak vermektedir. Böylece, çok aşamalı bir onay işlemi, detaylar gözardı edilmeksizin, kolay ve hızlı bir şekilde sona erdirilebilmekte; yöneticilerin iş akışlarını izleyebilmeleri, değiştirebilmeleri, aşamaları kontrol edebilmeleri, işlemlerin başlama ve bitiş tarihlerini düzenleyebilmeleri LAN/WAN/Internet/Intranet üzerinden sağlanabilmektedir. EDY sistemlerinin **COLD** modülü ise, mainframe'ler tarafından periyodik olarak üretilen bilgisayar kaynaklı elektronik raporların, kağıt ortamına geçirilmeksizin, doğrudan Optik Depolama Birimlerine (Optical Jukebox) alınarak arşivlenmesini sağlamaktadır. Bankalarda üretilen kağıt dokümanların birçoğu, bilindiği gibi, "zebra kağıt" olarak adlandırılan mainframe raporlarıdır. Bu raporlar, elektronik ortamda üretildikleri halde çoğu zaman zebra kağıtlara bastırılmakta, depolarda kağıt ortamında saklanmaktadır. Türkiye'deki orta ölçekli bir kurumsal bankada yapılan analiz, bu bankanın her yıl yaklaşık olarak 9 milyon sayfa zebra kağıt kullandığını göstermektedir. Bu ise, bankanın tüm kağıt doküman üretiminin yaklaşık olarak %30'udur. Yeniden yapılanma kararı alan bir banka, bir yıl içerisinde, EDY sistemine geçişle birlikte, yıllık 9 milyon sayfalık mainframe raporunu, zebra kağıda bastırmadan, doğrudan elektronik ortama alabilecek; rapor üzerinde, satır bazında bil

arayabilecek; daha önce tanımlanan parametrelerle filtreleme yapabilecek; raporlar üzerine açıklayıcı notlar, resimler, tablolar ekleyebilecek; kişiye özel mühür, silgi, kalem, markör kullanabilecektir.

2.6.1 Entegre doküman yönetim sistemleri ile gelen büyük değişim

Entegre doküman yönetim sistemleri, beraberlerinde büyük kurumsal değişimler getirmekte ve kağıda dayalı eski alışkanlıkları ortadan kaldırmaktadırlar. Böylece bankalara; üretkenlik, verim, müşteri memnuniyeti, karlılık artışı ve maliyet kontrolü gibi büyük stratejik kazançlar sağlamaktadır.

Banka müşterileri, işlemlerini en hızlı ve güvenilir şekilde yapan bankaları tercih etmektedirler. Bankalar da bu nedenle en yeni teknolojilere sürekli yatırım yapma zorunluluğu içerisindeyler. Gerçekten de müşteriler, kendilerine en kaliteli ve hızlı hizmeti veren, bilgilerine en hızlı ulaşan ve telefonun diğer ucundaki müşterisine anında cevap verebilen bankaları tercih etmektedirler.

Doküman yönetim sistemleri, bu bankaların, müşterilerine doğru bilgiyi, doğru zamanda, doğru biçimde verilebilmesine, hizmet kalitesini ve müşteri memnuniyetini arttırılabilmesine olanak tanımıştır. Sistem, yapısal olarak tanımlanmış işlemleri (müşteri fakslarının değerlendirilmesi, kredi kartı başvuru işlemleri v.b.) mümkün olduğunca insan müdahalesini en aza indirerek otomatik olarak yapmakta, dolayısı ile işlemleri hızlandırmakta, işlem hacmini, müşteri memnuniyetini ve verimi arttırmakta, bilgi paylaşımına olanak tanıyarak yöneticilerin daha etkili kararlar alabilmelerini sağlamaktadır. İşlemlerdeki insan müdahalesi en aza indirildiğinden, insandan kaynaklanan hatalar azaltılabilmekte ve çalışanlar, yanlış dosyalanan veya kaybolan dokümanları aramak yerine kendi işlerine konsantre olabilmektedirler. Ayrıca, her tür dokümanın elektronik ortamda saklanması sonucunda, dokümanların maruz kaldığı; kaybolma, yıpranma, yanlış birimlere gitme veya yanlış dosyalara girme gibi riskler ortadan kaldırılabilir. Doküman yönetim sistemleri, kağıt ortamında işlem yapmaktan kaynaklanan, dosyalama ve saklama ile ilişkili olan işgücü, kırtasiye, fiziksel depolama alanı v.b. kurumsal giderleri de azaltmakta, bankaların karlılığını arttırmaktadır.

2.6.2 Entegre doküman yönetim sistemi seçimi

Bankalar, tanım gereği, riskten kaçınan kurumlardır. Her yeni sistem gibi EDY sistemleri de beraberlerinde bazı riskler getirmektedirler. Bir banka için hayati değeri olan dokümanların

saklandığı sistemin bir an için bile durması düşünülemez. Bu nedenle bankaların, EDY sistemi seçiminde, aşağıda açıklanan maddeleri detaylı olarak sorgulaması gerekmektedir.

2.6.2.1 Güvenilirlik ve güvenlik

EDY sistemleri için güvenlik, sorgulanması gereken en önemli kriterlerden birisidir. Bir sistemin kuruma kazandırılmasından önce; sistemin aksaklıklara veya tamamen devre dışı kalmaya ne kadar eğilimli olduğu, böyle bir durumda tekrar eski konumuna gelebilmesi için ne kadar zaman ve para gerektirebileceği, sistemin devre dışı kaldığı zaman süresince oluşacak kayıplar (kaybolan müşteriler, işgücü kaybı v.b.) detaylı olarak sorgulanmalıdır. Sistem ayrıca, kendi içerisinde de bir güvenlik mekanizmasına sahip olmalı, doküman bazında yetkilendirme yapabilmeye olanak tanınmalıdır. Böylece, yetkisi olmayan personelin, kurumun kritik dokümanlarına ulaşması ve bilinçli veya bilinçsiz olarak bu dokümanlara zarar vermesi önlenmiş olur.

2.6.2.2 Felaket Senaryosu

EDY sistemleri ne kadar güvenilir olursa olsun, sistemin sağlıklı bir şekilde işlemeye devam edebilmesi için yedekleme şarttır. Yedekleme ile bu tür sistemlerin güvenliği önemli ölçüde artırılabilmekte, kayıplar kolayca yerine koyulabilmekte ve kurumdaki işlemler için hayati derecede önem taşıyan sistemin kesintisiz olarak çalışmaya devam etmesi sağlanmaktadır. Bundan dolayı, EDY sisteminin gelişmiş felaket senaryosu seçenekleri olması gerekmektedir.

2.6.2.3 Performans

EDY sistemleri birçok modülden oluşmakta ve bankanın altyapısı üzerinde çalışmaktadır. Bu nedenle, doküman tarayıcılardan işletim sistemlerine kadar her bir elemanın uyum içerisinde çalışması gerekmektedir. Ayrıca, her bir birim gelecekteki kapasiteyi kaldırabilecek şekilde planlanarak sisteme dahil edilmelidir. Kurumlarda genellikle, pilot projelerdeki doküman yoğunluğuna göre satın alınan birimler, proje genişledikçe ve kapasite arttıkça yetersiz kalabilmekte ve diğer birimlerin de çalışmasını yavaşlatmakta veya engellemektedir.

2.6.2.4 Mainframe entegrasyonu

EDY sistemlerini kullanmaya aday birçok banka, halen birçok uygulamalarını mainframe üzerinde çalıştırmaktadır. Öte yandan, birçok EDY sistemi, client/server yapısı gerektirmektedir. EDY sistemleri gerektirdiği için, bir bankanın, mainframe yapısından client/server yapısına geçmesi veya iki yapıyı paralel çalıştırması oldukça güç ve pahalı bir

işlemdir. Bu nedenle, EDY sistemlerinin mainframe yapısını da destekler konumda olması tercih edilmelidir.

2.6.2.5 Uygulama entegrasyonu

EDY sistemleri, kurumdaki kullanıcıların dokümanlarla ilgili her türlü ihtiyacına cevap verebilmelidir. Bu nedenle, gerek kurumda daha önceden varolan sistemlerle (legacy systems), gerek faks, OCR v.b. sistemlerle kolayca entegre olabilmeli, bu sistemlerle doküman alışverişi yapabilmelidir. Günümüz ofis ortamlarında bu bir zorunluluk haline gelmektedir. EDY sistemlerinde, örneğin; çok kartlı/çok hatlı faks hizmet birimleri tarafından alınan bir faks, bir uygulamaya otomatik olarak yönlendirilmekte, form tanıma (form recognition) kullanan bu uygulama ile her bir alandaki bilgiler okunmakta ve okunan bilgiler bir sonraki uygulamaya otomatik olarak geçirilmekte veya bir veritabanında saklanmaktadır. Uygulama tarafından otomatik olarak tanımlanamayan alanları içeren fakslar ise bir operatöre yönlendirilmekte ve işlem manuel olarak yapılmaktadır. Böylece kullanıcı müdahalesi en aza indirilmekte ve kuruma büyük kazançlar sağlanabilmektedir. Bundan dolayı, EDY sistemlerinin, diğer uygulamalarla, verimli ve güvenilir bir biçimde entegre edilebilmesi şarttır.

2.6.2.6 İşakışı (workflow) entegrasyonu

EDY sistemlerinin tamamına yakını, dokümanlardan bilgilerin otomatik olarak bulunup çıkarılması, dokümanların yönlendirilmesi, öncelik sıralamasına koyulması gibi fonksiyonlara ihtiyaç duydukları için iş-akışı uygulamaları kullanmak durumundadırlar. İş-akışı uygulaması üzerinden yeni gelen bir doküman ile ilgili, arşivdeki dokümanların da aynı anda kullanıcının ekranına getirilmesi, bankaların çağrı merkezi uygulamalarında kullanılan basit bir entegrasyon örneğidir. İş-akışı uygulamalarının bir diğer amacı ise iş yönetimi ve kontrolüdür. İş-akışı uygulamaları ile her bir işlemin hangi aşamada olduğu, nerede beklediği veya takıldığı, hangi birimin veya kullanıcının ne kadar zamanda işini bitirdiği, bir yönetici veya yöneticiler grubu tarafından rahatlıkla takip edilebilir. Böylece, kurumun verimliliği ve performansı, yöneticilerin ise iş üzerindeki etkinlikleri arttırılabilir. Bu nedenle, EDY sistemlerinin, iş-akışı sistemleriyle entegre edilebilmesi kurum açısından büyük önem taşımaktadır.

2.7 Sabit Disk Maliyetleri

Elektronik doküman yönetim sistemlerinde, dokümanları elektronik ortamda saklamak için başlıca yöntemlerden bir tanesi de dokümanları manyetik (sabit) disklerde tutmaktır.

Manyetik disklerin maliyeti konusu üzerinde yapılan bir araştırma da aşağıdaki çarpıcı tabloya rastlanmıştır.

Çizelge 2.1 1992-2010 yılları arası sabit disk maliyet tahmini

(<http://archivebuilders.com/whitepapers/22011.doc>)

Yıllık Düşüş 45%	1 GigaByte 1,000 Mbytes	1000 GB = 1 TeraByte'lık Bilgi Saklama Maliyeti (2,000 dosya dolabı) (Mektup zarfı boyutlarında 20 Milyon doküman için yeterli yer)				
	Bilgi Saklama Maliyeti	Non-FC/SCSI	Non-FC/SCSI	SAN	SCSI/FC SAN/PC	
	(ABD Doları)	PC Disk	PC Disk	FC Disk	Name Brand	Mainframe
	(2 Dosya dolabı)	No Online	Software RAID	FC Fabric	Fault Awareness	
	Yıl	Redundancy	Redundancy	Hardware RAID	Hardware RAID	
		1 X	2 X	4 X	8 X	12 X
1992	1,000.00	1,000,000.00	2,000,000.00	4,000,000.00	8,000,000.00	12,000,000.00
1993	550.00	550,000.00	1,100,000.00	2,200,000.00	4,400,000.00	6,600,000.00
1994	302.50	302,500.00	605,000.00	1,210,000.00	2,420,000.00	3,630,000.00
1995	166.38	166,375.00	332,750.00	665,500.00	1,331,000.00	1,996,500.00
1996	91.51	91,506.25	183,012.50	366,025.00	732,050.00	1,098,075.00
1997	50.33	50,328.44	100,656.88	201,313.75	402,627.50	603,941.25
1998	27.68	27,680.64	55,361.28	110,722.56	221,445.13	332,167.69
1999	15.22	15,224.35	30,448.70	60,897.41	121,794.82	182,692.23
2000	8.37	8,373.39	16,746.79	33,493.58	66,987.15	100,480.73
2001	4.61	4,605.37	9,210.73	18,421.47	36,842.93	55,264.40
2002	2.53	2,532.95	5,065.90	10,131.81	20,263.61	30,395.42
2003	1.39	1,393.12	2,786.25	5,572.49	11,144.99	16,717.48
2004	0.77	766.22	1,532.44	3,064.87	6,129.74	9,194.61
2005	0.42	421.42	842.84	1,685.68	3,371.36	5,057.04
2006	0.23	231.78	463.56	927.12	1,854.25	2,781.37
2007	0.13	127.48	254.96	509.92	1,019.84	1,529.75
2008	0.07	70.11	140.23	280.45	560.91	841.36
2009	0.04	38.56	77.13	154.25	308.50	462.75
2010	0.02	21.21	42.42	84.84	169.68	254.51

Çizelge 2.1, 1992 yılından 2010 yılına kadar 1 GB ve 1 TB için maliyet hesaplarını göstermektedir. Çizelge 1992'den 2000'e kadar her yıl %45'lik düşüşü baz alarak 2010 yılına kadar olası doküman saklama maliyetlerini ortaya koymaktadır.

Tabloda görünen maliyetteki bu düşüş IBM'in disk depolama yoğunluğundaki artışlar göz önüne alınarak ortaya koyulmuştur. IBM'in her yıl disk depo kapasitelerindeki bu %60'luk rakam göz önüne alınırsa, herhangi bir zaman diliminden bir yıl sonra aynı maliyetle 1.6 kat daha fazla bilgi depolanabilecektir. Bu fiyatlarda her yıl %37,5 lık bir düşüşü de beraberinde getirecektir.

IBM 29 Aralık 1997'de, disk maliyetlerindeki düşüşün devam edeceğini açıkladı. Neredeyse 2 yıl geçmişti ki IBM 4 Ekim 1999'de disk kapasitelerindeki artışın %60'lardan %100'lere çıkacağını anons etti ve 15 Ekim 1999'da IBM 73 GigaByte'lık, 2GB/saniye düzeyinde seri fiber bir kanala sahip disk yaptığını bildirdi. IBM firması disk kapasiteleri üzerindeki araştırmalarını daha da hızlandırarak 18 Mayıs 2001 tarihinde 48 GigaByte'lık kişisel bilgisayarlar da kullanılabilecek sabit diskleri piyasaya sürdü ve 2003'de bu kapasitenin 200 GB olması bekleniyor. (<http://archivebuilders.com/whitepapers/22011.doc>)

Kısaca, disklerin kapasiteleri artarken fiyatları düşüyor. Bu sebeple disklerde saklama maliyeti giderek azalıyor. Bundan 4-5 sene sonra bu maliyeti her hangi bir kullanıcı bile karşılayabilecek durumda olacak, firmalar ise elektronik doküman yönetim sistemlerinin disk maliyetlerini hesaplarına bile katmıyor olacaklar.

2.8 Doküman Yönetim Sistemi Genel Kavramları

Doküman yönetim sistemlerinin kendine has kavramları vardır. Bu kavramların açıklanması sistemin anlaşılması açısından önem taşımaktadır.

Sistemde yer alan belli başlı kavramlar,

- Anahtar Alanları
- İndeksleme
- İndeksleme Sınıfları
- Klasörler
- Çalışma Sepetleri(Listeleri)
- Giriş Kontrol Listeleri
- Ayrıcalıklar

2.8.1 Anahtar alanları

Bir dokümanın indekslenmesinde kullanılacak veritabanı alanlarından herbirine anahtar alanı adı verilir. Müşteri_Ad, Müşteri_Soyad örnek birer anahtar alandırlar. Minimum ve maksimum birer uzunlukları vardır. Bunlar kütüphane sunucusunun veritabanında bir tabloda tutulurlar. Bir indeks sınıfı bu anahtar alanlar topluluğunun bir alt kümesinden oluşturulur.

2.8.2 İndeksleme

Bir dokümanı indekslemek demek dokümanla ilgili onu tanımlayan bazı bilgileri ilgili veritabanı tablosuna yazmak demektir. Mesela elinizde müşterinizin bir evrakı, MUSTERİ adında bir indeks sınıfı ve bu indeks sınıfının içinde SicilNo, Musteri_Ad,Musteri_Soyad alanları varsa, dokümanın bu bilgilerle beraber tabloya kaydedilmesi olayına indeksleme adı verilir.

2.8.3 İndeksleme sınıfları

İndeksleme sınıfları bir dokümanın bilgilerinin indekslenmesinde kullanılmak için dinamik olarak oluşturulan veritabanı tablolarıdır. Mesela bir müşteri dokümanının indekslenmesi için Ad, Soyad, SicilNo, Telefon, Adres gibi bilgileri tutabilecek bir tabloya ihtiyaç olabilir. Bu durumda sisteme söylenmesi gereken, tanımlı olan anahtar alanlarından bu beşini birleştirerek bir tablo oluşturmaktır. Bu tablo ayrı bir indeks sınıfı olarak adlandırılır.

2.8.4 Klasörler

Klasörler, diğer klasörleri veya dokümanları içerisinde taşıyabilen kataloglardır. Klasörlerde sistemde dokümanlar gibi indekslenebilir veya çalışma listelerine yönlendirebilirler.

2.8.5 Çalışma listeleri

Çalışma listeleri dokümanları veya klasörleri iş akışına sokabileceğimiz iş kuyruklarıdır.

2.8.6 Giriş kontrol listeleri

Giriş Kontrol Listeleri bir kullanıcının bir indeks sınıfına veya bir çalışma sepetine hangi haklarla ve hangi ayrıcalıklarla gireceğini belirleyen listelerdir.

2.8.7 Ayrıcalıklar

Ayrıcalıklar bir indeks sınıfında veya bir çalışma listesinde yazma, okuma, ekleme ve diğer

bazı hakları barındıran gruplardır.



3. TASARIM

Dokümanların elektronik ortama alınması, paylaşımı, şirket içinde dolaşımı ve yönetimi birden fazla kullanıcı bir doküman yönetim sisteminin tasarlanmasını birlikte getirir. Bu birden fazla modülden oluşan bir yapıdır. Bu bölümde bu modüllerin hangi ihtiyaçlardan dolayı ortaya çıktıkları ve ne amaca hizmet ettikleri detaylı olarak anlatılacaktır.

3.1 Sistemde Olması Gereken Modüller

Doküman yönetim sistemleri tüm doküman tiplerini elektronik ortamda saklayabilen, yönetebilen, isteyen kişinin hizmetine sunabilen, güvenliğini sağlayabilen ve bir iş akışına sokabilen sistemlerdir.

Doküman yönetim sistemi fiziksel bir kütüphane yapısının elektronik ortamda çok daha detaylı bir şekilde tasarlanmış halidir. Doküman yönetim sistemi içerisinde yüzlerce kişinin çalışabildiği, kendi iç güvenliğini kendisi sağlayan, çok kapsamlı bir elektronik kütüphanenin yönetilmesidir.

Ancak tasarlanan elektronik kütüphane sistemi günümüzdeki fiziksel kütüphaneler gibi sadece metin dosyalar için hizmet vermemelidir. Bu sistem bir Word dokümanı, bir Excel dokümanı, bir TIF görüntü dosyası, bir MPEG video görüntüsü, bir WAV ses dosyası yönetebilmelidir. Hatta bir doküman nesnesi mantığı geliştirip, bir doküman nesnesi içinde birden fazla doküman tipi yönetebilmelidir. Bu alt doküman parçalarını versiyonlama şansı da sağlamalıdır. Tasarlanan sistem de bunların hepsine genel olarak doküman adını verdik. Bu dokümanlar bir elektronik kütüphaneye yerleştirildi. İlk yapılması gereken işlerden bir tanesi dokümanları yerlerine yerleştirmek ve bunları yerleştirdiğimiz yerleri sisteme kaydetmektir. Çünkü dokümanları aradığımızda çok kolay ve hızlı bir şekilde bulmamız gerekir.

Kütüphanelerde dokümanların bir araya getirilip tutuldukları nesneler klasörlerdir. Bu durumda tasarlanacak elektronik kütüphanede klasör mantığı olmalıdır. Klasörler ve içinde bulundurduğu dokümanların detayları sistemin kütüphane yapısı içinde tutulmalıdır. Tabi tasarlanacak elektronik sistemde klasörlerin içinde sadece kağıtlardan oluşmuş nesneler olmayacaktır.

Ayrıca aynı doküman birden fazla klasöre bağlanabilmelidir. Klasörler bir değil tüm doküman tiplerini tutmak için tasarlanmalıdırlar. Yani bir klasör içinde bir TIF imaj dosyasıda, bir MP3 müzik dosyasıda, bir MPEG görüntü dosyası da olabilir. Hatta bir klasör başka bir klasör içerebilmelidir.

Elektronik kütüphanenin yani doküman yönetim sistemimizin yapması gereken bir başka iş, sistemin içerdiği dokümanlara erişecek kişiler için güvenliği sağlamasıdır. Güvenlik en verimli ve en doğru şekilde gerçekleştirilmelidir. Günümüz fiziksel kütüphanelerinde her isteyen her istediği dokümana ulaşamıyorsa, tasarlanacak elektronik sistemde de her kullanıcı istediği her dokümana ulaşamamalıdır. Bundan dolayı ayrıcalık setleri tasarlanmalıdır. Her ayrıcalık bir güvenlik birimidir. Mesela bir dokümanın okunması bir ayrıcalıktır. Dokümanın güncellenmesi başka bir ayrıcalıktır. Bunları bir araya getirerek ayrıcalık setleri oluşturulmalıdır. Ayrıca bu ayrıcalık setlerinin bir kullanıcıyla beraber tanımlanacağı giriş kontrol listeleri de tasarlanmalıdır. Giriş kontrol listeleri bir kullanıcının kütüphane sistemindeki bir nesneye (indeks sınıfı veya çalışma listesi) hangi ayrıcalıklarla erişeceğini göstermelidir. Bu listeler sisteme güvenliğin en esnek ve ideal şekilde getirilmesini sağlar.

Elektronik kütüphanede bulundurulanan dokümanları ve klasörleri tanımlayabilecek bilgilere ihtiyaç duyulabilir. Yani elimizdeki bir dokümanın ne olduğunu, ne için kullanıldığını, en son ne zaman güncellendiğini, en son kim tarafından okunduğunu bilmek isteyebiliriz. Aynı şey elektronik olarak tasarlayacağımız klasörler için de geçerli olabilir. Bu işleme doküman yönetim sistemlerinde indeksleme adı verilir. Diğer taraftan bir dokümanı indekslemek için kullanılacak bilgiler bir başka dokümanı indekslemek için kullanılacak bilgilerle aynı olmayabilir. Dışarıdan gelen bir faksı indekslemek için nereden geldiği ve ne zaman geldiği yeterli bir bilgi olurken, bir müşteri klasörünü indekslemek için müşteri sicil numarası, müşteri adı ve soyadı bilgileri yeterli olabilir. Bu nedenle dokümanları veya klasörleri farklı indeks listelerinde saklamamız gerekecektir. Bu farklı indeks sınıflarının tasarlanacak sistemde olması gerekmektedir.

Doküman veya klasörler bir iş akışında kullanılabilirler. Doküman yönetim sistemi kullanmayan yerlerde fiziksel klasör veya dokümanlarla iş akışını gerçekleştirmek çok zaman alan zahmetli bir iştir. Ayrıca nesnelerin kaybolma riski de vardır. Güvenlik ise yok denecek kadar azdır. O zaman doküman yönetim sistemlerinde mutlaka iş akışının sistemin tasarımına yerleştirilmesi gerekmektedir. Çalışma listeleri tasarlanması bu problemi çözebilecektir.

Bunun yanında sistemde kimlerin çalışacağı belirlenmeli ve doküman yönetim sistemine tanıtılmalıdır. Elektronik kütüphane içinde hangi kullanıcıların sisteme giriş yapabilecekleri bilgisi tutulmalıdır. Ayrıca kullanıcı yönetiminin kolay olabilmesi amacıyla bazı kullanıcı grupları yapısına da ihtiyaç duyulmaktadır.

Bir diğer mesele dokümanların fiziksel olarak elektronik ortamda saklanması için bazı dosya

sunuculara ve bunların sabit disk bilgilerine ihtiyaç vardır. Dosya sunucuları bilgilerinin de ana sistemde tutulması gerekmektedir.

Tüm bunları yapabilecek modül **kütüphane sunucusu modülü** olarak adlandırılacaktır. Bunlar sistemin katalog bilgileridir ve merkezi bir veritabanında tutulmalıdırlar.

Dokümanların elektronik ortamda saklanması için fiziksel disk depolarına ihtiyaç duyulur. Bu depolar dosya sunuculardır. Asli görevleri dokümanları, üzerlerinde bulundurdukları sabit disklerde saklamak ve istenildiğinde bunları isteyen istemcilere yollamaktır. Bu sunucular **doküman sunucular** olarak adlandırılırlar.

Yedekleme süresini kısa tutabilmek ve yedekleme riskini en aza indirmek için doküman sunucuların sabit disklerini mantıksal parçalara ayırma olanağının sağlanması gerekmektedir. Bunlardan herbirine disk parçası adını vermekteyiz. Mesela, üzerinde 100GB lık bir disk bulunan doküman sunucunun diskleri mantıksal olarak 5 parçaya ayrıştırılıp 20GB lık disk parçaları olarak kullanılabilir. Özellikle tasarlanacak sistemde aynı anda sadece bir disk parçasının aktif olarak işaretlenip, fiziksel olarak sadece bu disk parçası üzerinde güncelleme yapılması çok önemlidir. Sistem disk parçaları sistemin kütüphane sunucu veritabanına tanımlandığında, sistem bir disk parçasını aktif olarak işaretlemeli ve sisteme yeni gelen dokümanları bu disk parçası üzerine kaydetmelidir. Bu disk parçası dolduğunda yeni bir disk parçası seçilmeli ve yeni doküman kaydetmeler bu disk parçası üzerine yapılmalıdır. Ancak daha önce dolu olan disk üzerindeki bir doküman güncellenmek istendiğinde, dosyanın güncellendikten sonraki görüntüsü sistemde o anda aktif olan disk parçasına kaydedilmelidir. Böylece eski disk parçasında herhangi bir değişiklik olmaz. Bu durum sadece o anki aktif disk parçası üzerinde yedekleme yapma kolaylığını sağlar.

Kütüphane sunucu üzerindeki tüm konfigürasyonların yapılabilmesi için bir **yönetici uygulamasına** ihtiyaç vardır. Kullanıcıların tanımlanması, gruplandırılması, ayrıcalık setlerinin oluşturulması, giriş kontrol listelerinin tanımlanması, sistem konfigürasyon parametrelerinin kurulması, çalışma listelerinin oluşturulması, indeks sınıflarının yaratılması gibi tüm tanımlamalar bu yönetici program tarafından yapılabilir.

Sistemdeki bazı işlerin otomatikleştirilmesinde fayda vardır. Doküman sunucuların üzerlerindeki disk parça bilgilerinin takip edilmesi, disk parçasının alanı dolduğunda otomatik olarak başka bir disk parçasının seçilmesi, doküman sunucu bağlantılarının kontrol edilmesi, gerektiğinde bu durumların sistem yöneticisine mailenmesi gibi işler otomatik yapılabilecek işlerdir ve bu işleri yapan modül **SMS sunucu** adıyla adlandırılır.

Tüm sistemi inşa ettikten sonra bu sistemi kullanacak, tüm sistemin mantığını üzerinde bulunduran bir uygulamaya ihtiyaç duyarız. Bu uygulama sisteme doküman ekleyebilmeli, klasör yaratabilmeli, doküman veya klasörleri arayıp bulabilmeli, bunları başka klasörlere yerleştirebilmeli, çalışma listelerine yönlendirebilmeli, çalışma listelerinden çıkartabilmelidir. Ayrıca dokümanları açabilmeli, hem indeks bilgisini hem de dokümanın kendisini güncelleyebilmelidir. Bütün bu işlemleri aynı anda yüzlerce kullanıcıyla yapabilmelidir. Yani bir yönetim mekanizmasına sahip olmalıdır. Güncelleme işleri için bir senkronlama mekanizması geliştirmelidir.

Ayrıca bu istemci uygulama öyle bir uygulama olmalıdır ki diğer tüm uygulama geliştirme araçları bu uygulamayı kullanarak kendi doküman yönetim istemci uygulamalarını gerçekleştirebilsinler. Bu durumda istemci uygulaması Windows işletim sisteminde çalışmalıdır. İstemci uygulama bir OLE objesi olmalı ve bütün uygulamalar tarafından kullanılabilmelidir.

Bu istemci uygulaması içerisinde ana iki modül yer almalıdır. Bir modül kütüphane sunucusu ile çalışmalı, diğer modül ise doküman sunucularla doküman alış verişinde bulunmalıdır. Yani bir modül kütüphane sunucusundaki doküman check-out, doküman check-in, çalışma listesi listeleme, klasör veya doküman yaratma, silme, güncelleme gibi tüm kütüphane sunucu ile ilgili işleri gerçekleştirebilmeli, diğer bir modül ise doküman sunucu üzerinde doküman saklama veya doküman sunucudan bir doküman getirme işlerini halledebilmelidir. Biz Windows tarafında çalışabilecek bu uygulamaya **standard istemci uygulaması veya nesnesi** adını veriyoruz.

Özetlersek, sistemde istemciler, kütüphane sunucusu ve doküman sunucular ana modülleri oluşturmalıdır. Diğer taraftan yönetici program kütüphane sunucusu üzerinde yapılacak tanımlamalara olanak sağlarken, SMS sunucuda sistemde tüm dokümanların tutulduğu sabit disk yönetimiyle uğraşacaktır. Yani ana modül bazında 3, detaylı modül bazında 5 modüle sahip bir sistem tasarlanacaktır.

3.2 Tasarlanan Sistem Yapısı

Tasarlanan Doküman Yönetim Sistemi 3 ana modülden oluşur. (Şekil 1.1)

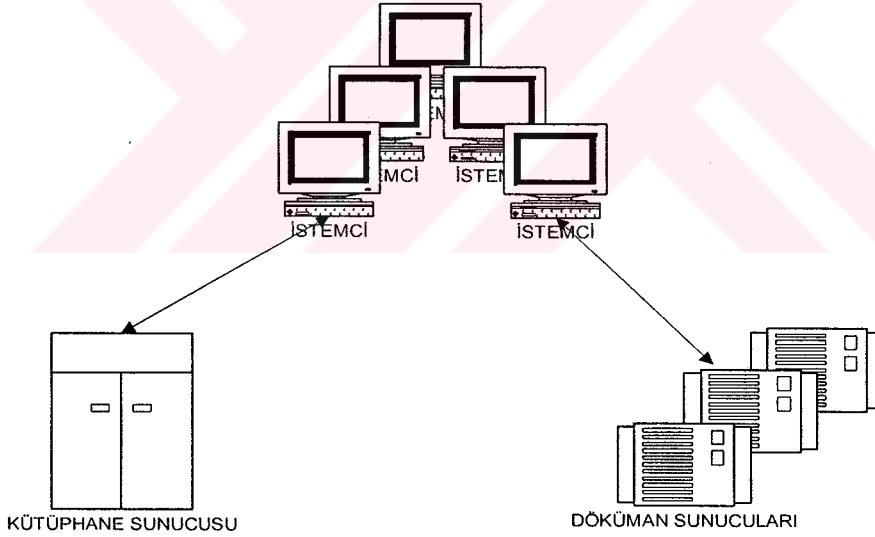
1. İstemciler
2. Kütüphane Sunucusu

3. Doküman Sunucusu

İstemciler sistemin kullanıcılarıdır. Kullanıcılar kütüphane sunucusunun veritabanında tanımlıdır. Bir şifre ile beraber kullanıcı adlarıyla sisteme logon olurlar. İstemciler kütüphane sunucusundan dokümanların yerlerini öğrenir ve doküman sunucusundan da istedikleri dokümanı alırlar. İstemciler, yetkileri elverdiği sürece, kütüphane sunucusunda dokümanın bilgilerini indekslemek koşuluyla, doküman sunucusuna yeni dokümanlarda yerleştirebilirler. İstemci uygulamaları Windows NT ve Windows 2000 üzerinde çalışabilirler.

Kütüphane sunucusu veritabanında tüm sistem bilgilerini kataloglayan makinedir. Üzerinde bir veritabanı uygulaması bulundurulur. Kullanıcılar, kodlanmış şekilde şifreler, indeks sınıfları, çalışma sepetleri, ayrıcalıklar, giriş kontrol listeleri, imaj sunucu bilgileri tanımlıdır. Tüm doküman yönetim sisteminde sadece 1 tane kütüphane sunucusu bulunmak zorundadır.

Doküman sunucusunda ise tüm imajlar tutulur. Tutulan doküman herhangi bir doküman formatında olabilir.. Bunların herbirine sistemde doküman adı verilir. Doküman yönetim sisteminde birden fazla doküman sunucusu bulunabilir.



Şekil 3.1 Sistemin basit modül şeması.

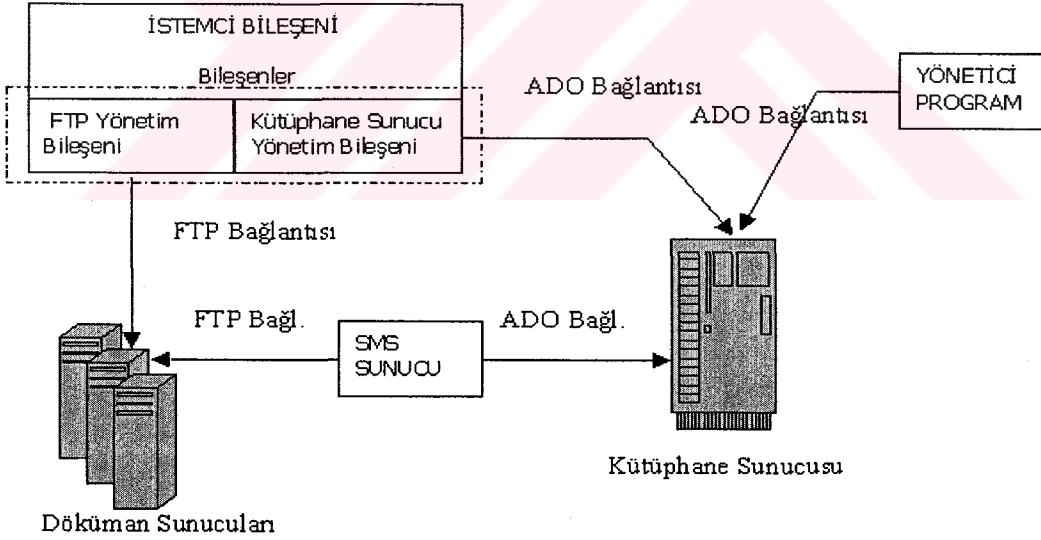
3.3 Sistem Mimarisi

Doküman yönetim sistemi temel olarak 5 bileşenin bir araya gelmesinden oluşur. Bu beş temel bileşenin adları İstemci, Yönetici Program, SMS Sunucu, Doküman Sunucu ve Kütüphane sunucusudur.

İstemci bileşeni, Windows NT/2000 ortamında bulunan, tüm mantığını FTP yönetim bileşeni ve kütüphane sunucu yönetim bileşeni parçalarına yayan ve fonksiyonlarını dışa açabilen bir OLE nesnesidir.

İstemci bileşeni kütüphane sunucusu ile ADO (Microsoft ActiveX Data Objects) bağlantısı ile konuşur. Aslında bu işi yapan, istemci bileşenindeki fonksiyonların bir parçasını oluşturan kütüphane sunucu yönetim bileşenidir. İstemci bileşeni kütüphane bileşeni ile sistemdeki tüm nesnelerin tasnifi, yönetimi ve yerütülmesi için iletişim içindedir.

İstemcinin FTP yönetim bileşeni ve doküman sunucusu FTP protokolüyle anlaşılır. Doküman sunucuları, dokümanların saklanması ve/veya istemcilere doküman gönderilmesinde esas görevi yapan sistem parçasıdır. Doküman sunucusu üzerinde, bir File Transfer Protokolünü (FTP) destekleyen bir sunucunun çalışması gerekmektedir. İstemci bu FTP sunucusu ile arasındaki bağlantı sayesinde dokümanları doküman sunucudan alabilir veya doküman sunucuya yerleştirebilir. İstemci doküman sunucunun adını, bu doküman sunucuda çalışan FTP programa nasıl giriş yapabileceğini yine kütüphane sunucusunda bulunan bir tablodan öğrenir.



Şekil 3.2 Sistemin detaylı modül yapısı

Kütüphane sunucusu sistemin ana merkezidir. Tüm sistem bilgisi kütüphane sunucusunun veritabanındaki tablolarında tutulur. Kütüphane sunucusunun konfigürasyonunu yapan program yönetici programdır. Yönetici program ile kütüphane sunucusu yine ADO bağlantı üzerinden konuşurlar.

SMS Sunucu özellikle dokümanların saklanacağı disklerin yönetimi ve kontrolü için çok önemli bir uygulamadır. Bu uygulama kütüphane sunucusunda bulunan parametrelerle açılarak sistemin disk yönetimini ve doküman sunucuların ulaşılabilirlik ve ulaşılamazlık kontrollerini yaparak gerektiğinde sistem yöneticisini uyarır.

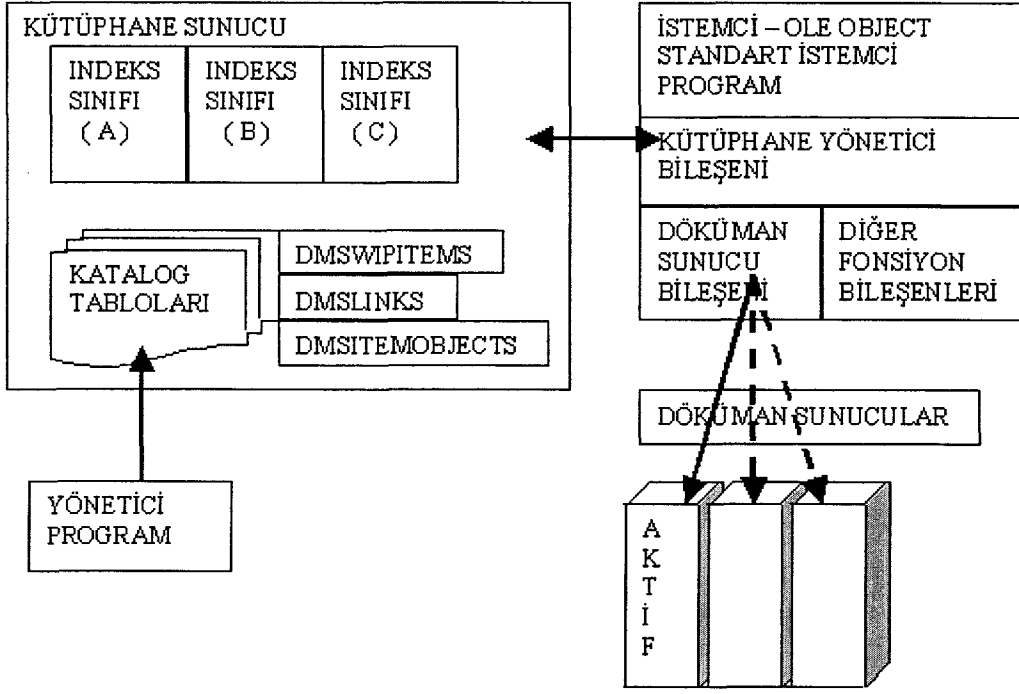
3.4 Sistemin Temel Yapısının Anlatılması

Doküman yönetim sistemi tüm mantığını neredeyse istemci üzerindeki bileşenlerine yükler. Bu akıllı nesne tüm sistem fonksiyonlarını üzerinde taşır.

Ancak sistemin çalışabilmesi için bazı konfigürasyonların yapılması şarttır. Kullanıcıların ve grupların tanımlanması, indeks sınıflarının ve çalışma listelerinin oluşturulması, ayrıcalık setlerinin ve giriş kontrol listelerinin belirlenmesi sistemin anlamlı bir hale gelmesi açısından önemlidir. Bunların tümü Yönetici Program tarafından yapılır. Bazı kavramların yerleşebilmesi için Yönetici Programın anlatıldığı bölümü okuyabilirsiniz.

Konfigürasyon yapıldıktan ve sistem çalışmaya başladıktan sonra asıl haberleşme istemci ve kütüphane sunucusu arasında gerçekleşmektedir. Doküman sunucular sadece doküman saklama veya dokümanı isteyen istemciye verme işinde görev üstlenirler. Bu işlem istemci bileşeni ile doküman sunucu üzerinde çalışan FTP sunucu arasında olur.

Kütüphane sunucusunun veritabanında katalog tablolarından üç tanesi dikkat çekmektedir. Bunlardan bir tanesi olan DMSITEMS tablosu tüm doküman nesnelerini ve bunların hangi indeks sınıflarında indekslendiğinin bilgisini tutar. İstemci bir dokümanı ekranda gösterecekse bu tablodan hangi indeks sınıfında indekslendiğini öğrenecektir.



Şekil 3.3 Sistemin çalışma şekli

DMSLINKS tablosu sadece klasörlerle diğer klasör veya dokümanlar arasındaki bağlantıları tutar. Bir klasörün içinde başka bir klasör de olabilir.

Önemli tablolardan bir diğeri ise çalışma listelerinin içinde bulundurdıkları nesneleri tutan DMSWIPITEMS tablosudur. Bu tablo tüm çalışma listelerinin detaylarını içinde barındırır. Çalışma listeleri bu tablodan listelenirler.

Tüm dokümanlar veya klasörler indeks sınıflarda indekslenirler. Yani bir klasör veya bir doküman mutlaka ama mutlaka bir indeks sınıfı tablosunda bulunmak zorundadır.

İstemci tüm sistemin mantığını uygulayan nesnedir. İstemci kütüphane sunucu ile çalışmak için kütüphane sunucu yönetici bileşenlerinin oluşturduğu veritabanı fonksiyonlarını, doküman sunucu ile çalışmak için doküman sunucu bileşeninin fonksiyonlarını kullanır. İstemcinin kendisi 32 bit Windows ortamında çalışabilen bir uygulamadır. Ancak kendisi bir OLE nesnesi olduğundan diğer tüm uygulamalar tarafından bir altyapı olarak kullanılabilirler. Yani anlatılan istemci uygulaması kullanılarak tamamen farklı bir önyüze sahip Visual Basic bir uygulama geliştirilebilir.

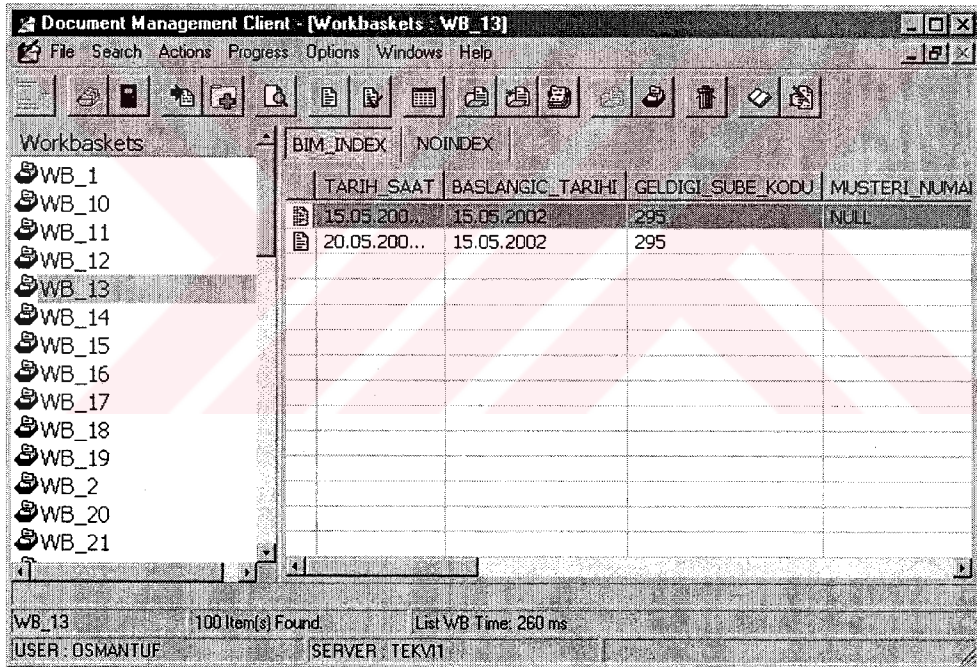
3.5 Ne Başarmak İstiyoruz?

Tezde tasarlanan sistemde başarılmak istenen temel düşünce, tüm dokümanlarımızı elektronik

ortamda düzenli bir şekilde saklayabilmemiz ve gerektiğinde firmamız içindeki bir iş akışına sokabilmemizdir.

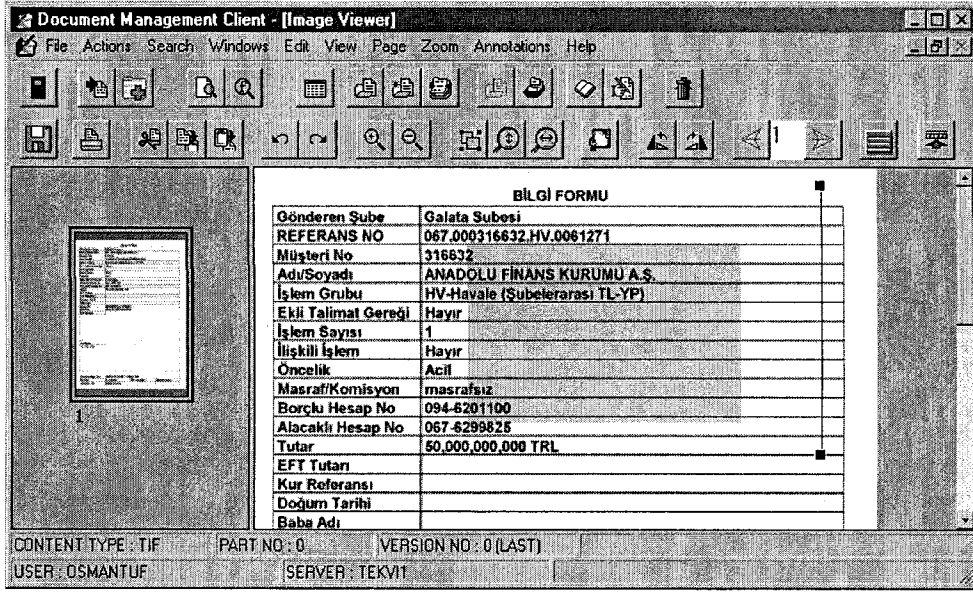
Sistemde doküman veya klasörlerin düzenli ve indeksli şekilde saklandıkları yerler indeks sınıflarıdır. Aşağıdaki şekilde tasarlanan doküman yönetim standard istemci uygulamasında BIM_INDEX ve NOINDEX iki ayrı indeks sınıflarıdır.

Görüldüğü gibi BIM_INDEX sınıfında indeks bilgileriyle beraber iki adet doküman vardır. Workbaskets penceresi altında ise Çalışma listeleri sıralanır. Bu çalışma listeleri iş akışı sağlar. İstenildiğinde bir doküman veya klasör bu listelere yönlendirilebilirler ve genellikle her çalışma listesinin bir sahibi vardır. Doküman veya klasör eğer çalışma listesinden çıkartılırsa indeks sınıfında indeksli bir şekilde sonsuza kadar kalabilir. Tabii ki istenildiğinde aranıp bulunabilir ve yeniden iş akışına sokulabilir.



Şekil 3.4 Dokümanların standart istemci uygulamasında indekslenmesi

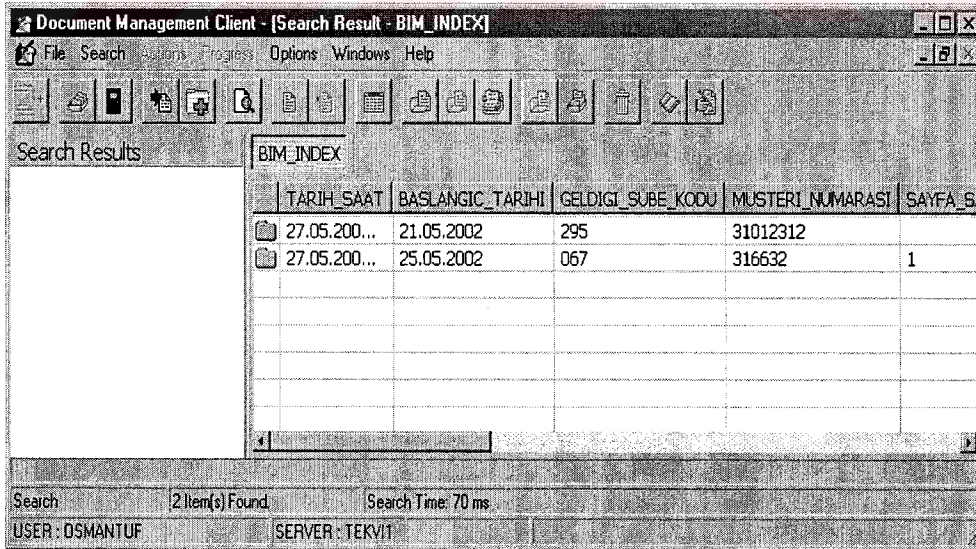
İstenildiğinde doküman açıp okunabilmeli istenildiği zaman doküman güncellenebilmelidir. Tabii ki bunu yapacak kişinin buna hakkının olması gerekir. Ayrıca güncellenebilme olayı doküman tipine de bağlıdır. Eğer doküman aşağıdaki şekilde gösterildiği gibi bir TIF imajı ise standard istemci uygulaması bunu açacak kadar yetenekli olmalıdır. Ancak bir MPEG gösterim dosyası sadece dinlenebilir! Doküman yönetim sisteminin windows için geçerli tüm doküman tiplerini yönetebilmek için tasarlanmış olması gerekir.



Şekil 3.5 Dokümanların standart istemci uygulamasında gösterilmesi

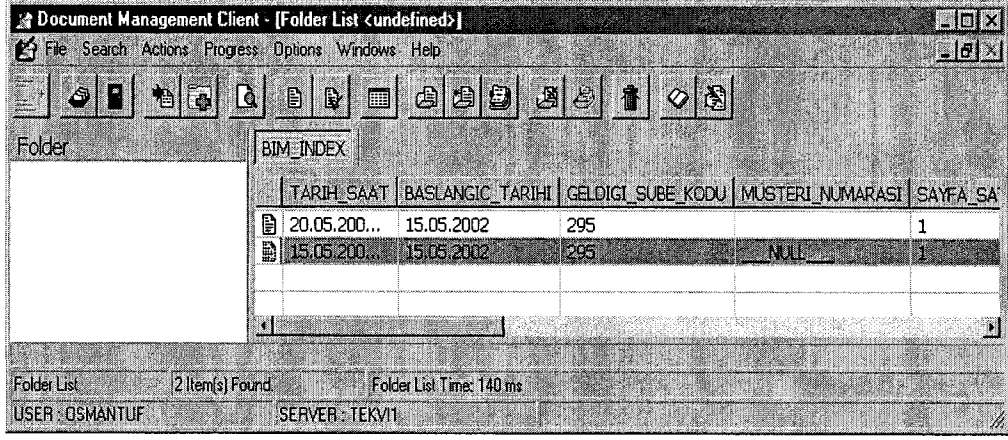
Aynı zamanda dokümanların indeks bilgilerinin değiştirilebilmesi, saklanması, versiyonlanması kolay yapılabilir olmalıdır. Ve bunları aynı anda yüzlerde kişiyle yapabilmek gerekir.

Klasörleri bulmak ve yönetmek ile ilgili işimiz basit olmalıdır. Klasörlere istediğimiz anda kolay bir şekilde ulaşmamız gerekir. Klasörleri de dokümanlar gibi indeksleyebilmemiz en uygun çözüm olacaktır.



Şekil 3.6 Klasörlerin standart istemci uygulamasında indekslenmesi

Bunların içindeki dokümanlara tek çift tıkla ulaşabilmemiz gerekmektedir.



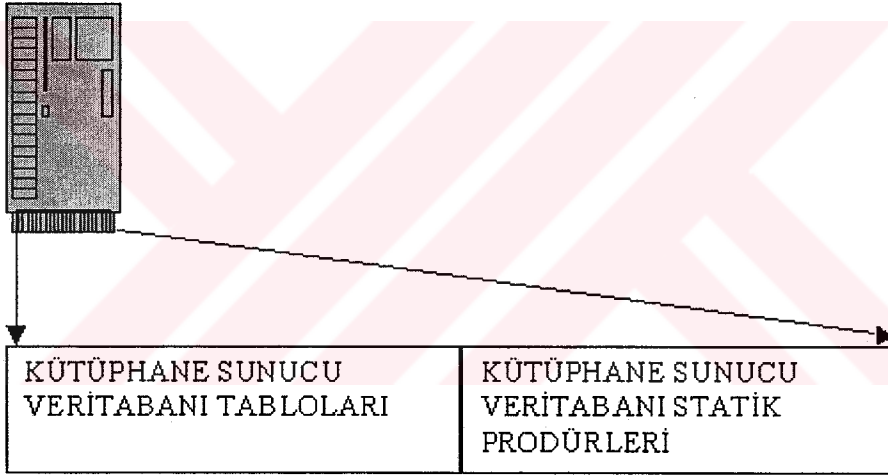
Şekil 3.7 Klasörleri standart istemci uygulamasında listelenmesi

Anlatılan tüm özelliklerin tasarlayacağımız sistem içinde mevcut olması gerekmektedir.

4. KÜTÜPHANE SUNUCU MİMARİSİ

Kütüphane sunucusu bir veritabanı uygulamasıdır. Windows ortamında bulunan bir istemci kütüphane sunucu veritabanına ADO bağlantısı ile ulaşır ve ya dinamik prosedür yada sunucu üzerindeki statik prosedürlerden birini çalıştırarak yapması gereken işi yapar. Bu bir seçim olabilir, ekleme olabilir veya güncelleme olabilir. Kütüphane sunucusu üzerindeki sistemsel konfigürasyonlarda yine bir yönetim uygulaması tarafından gerçekleştirilir. Bu yönetim uygulaması kütüphane sunucusuna ADO ile bağlanır. Ancak istemciden farklı olarak dinamik bir tablo yaratabilir veya yeni bir statik prosedür oluşturabilir. Yani kütüphane sunucusu ile çalışan modüllerin görevleri ve dolayısıyla yaptıkları işler birbirinden farklıdır.

Kütüphane sunucusunun mimarisini iki bölümde inceleyebiliriz. Bunlardan birincisi sunucuda bulunan tablolar, ikincisi ise sunucuda bu tablolar üzerinde işlem yapmak için yazılmış statik prosedürlerdir.



Şekil 4.1 Kütüphane sunucusu mimarisi

Kütüphane veritabanında esas olarak tablolar ve bu tablolar üzerinde değişik işlemler yapacak statik prosedürler bulunur. İstemciler işlemlerinin bir çoğunu bu statik prosedürleri kullanarak yaparlar.

Mesela bir dokümanı check-out (güncelleme için başkası tarafından kullanılamaz hale getirmek) etmek, check-in etmek (güncelleme için herkese serbest hale getirmek) , bir çalışma listesine yönlendirmek, bir çalışma listesinden çıkartmak gibi basit işlemler, veritabanı performansı dinamik procedürlere göre çok daha yüksek statik procedürler tarafından gerçekleştirilir.

Burada statik prosedürler ile dinamik prosedürlere değinmekte fayda vardır. Bir dinamik prosedür istemci tarafından oluşturulan bir sql cümlecığının sunucu üzerinde çalıştırılmasıdır. Bu dinamik cümlecik her çalıştırıldığında SQL Sunucu bu cümlecığın hem cümle kontrolünü yapmakta hemde çalışma şeklini belirlemek zorundadır. Ancak eğer bu sql ifade sunucu üzerinde bir statik sql olarak çalışacaksa, sunucu böyle bir statik prosedürü çalıştırmak için cümle kontrolü ve çalışma şekli için zaman kaybetmez. Bundan dolayı statik prosedürler dinamik prosedürlere göre daha avantajlıdır.

Şimdi sırayla veritabanındaki tablolar ve statik prosedürler anlatılacaktır

4.1 Kütüphane Veritabanı Tablolarının Oluşturulması

Kütüphane sunucusunun üzerinde kurulu veritabanı uygulamasında ürün kurulduğunda DMSDB adlı bir veritabanı yaratılır.

Bu veritabanı kütüphane sunucusunun tüm sistemin bilgilerini sakladığı ve yönettiği veri tabanıdır. Veritabanının içindeki tabloları 2 ana bölümde inceleyebiliriz.

1. Kurulum Sırasında Yaratılan Tablolar (Statik Varolan tablolar)
2. İndeks Sınıfları yaratıldıkça oluşturulan dinamik tablolar

Dokümanların veya klasörlerin indeks bilgilerini daha farklı tablo yapılarında tutabilmek için yeni indeks sınıflarları oluşturulurken, indeks sınıflarının veritabanındaki fiziksel karşılığı olan tablolarını oluşturmak gerekir. Bu işi yönetici program otomatik olarak yapar.

Kurulum sırasında yaratılan tabloların sayısı 22 tanedir. Ancak NOINDEX adıyla ayrıca bir indeks sınıfı açılarak SOC01000 adı verilir. Böylece toplam tablo sayısı 23 olur. İndeks sınıfları açıldıkça sırasıyla SOC01001, SOC01002 adıyla tablolar veritabanında oluşturulmaya başlanır.

Yönetim uygulama programı kullanılarak *bir indeks sınıfı yaratıldığında* sistem DMSMASTER tablosunun AVT_COUNT parametresinde tuttuğu sayıyı 1 arttırarak elde ettiği sayıyı SOC önekinin arkasına SOCXXXXX şeklinde ekleyerek elde ettiği isimdeki tabloyu otomatik olarak yaratacaktır. Bununla beraber iki tanede statik prosedür yaratılır. Bunun detaylarına dinamik prosedürlerin genel tanımında girilecektir.

SOCXXXXX tablosu ilk olarak tanımlanırken sadece ITEM_ID char(32) adlı bir kolon hızlı erişim amacıyla “veritabanında indeksli” olacak şekilde tanımlanır. Yönetici uygulaması

kullanılarak anahtar alanlar içinden bir alan eklendikçe yönetici uygulaması otomatik olarak bu tabloya bir kolon ekleyecektir. Böylece doküman veya klasörlerin indeks bilgilerinin tutulması için kullanılacak yeni bir veritabanı tablosu oluşturulacaktır.

Aşağıdaki çizelgede kurulum sırasında yaratılan NOINDEX indeks sınıfının alanları gösterilmiştir. NOINDEX dokümanların ilk yaratıldıkları indeks sınıfının adıdır.

Çizelge 4.1 NOINDEX tablo yapısı

KolonAdı	Veri Tipi
ITEM_ID	Char(32)
SOURCE	Varchar(50)
USERID	Varchar(50)
TIMESTAMP	Datetime

4.2 Kütüphane Sunucusunun Veritabanı Neler Olabilir?

Tasarlanan doküman yönetim sistemi üç temel bileşenden oluşmuştur. Bunlardan en önemli olanı tüm sistemin katalog bilgisini tutan kütüphane sunucusudur.

Kütüphane sunucusu içerisinde yirmiden fazla tablosu, yüzden fazla statik prosedürleriyle tam bir veritabanı uygulamasıdır.

Sistemdeki esas mantık istemci tarafındaki bileşenlerdedir. İstemciler kütüphane bileşenleri içinde tasarlanan fonksiyonları ile Windows ortamından ADO bağlantı yapısını kullanarak kütüphane sunucusundaki veritabanına ulaşırlar. Sistemin performansı kütüphane sunucusunun performansı ile birebir alakalı olduğu için seçilecek veritabanının performansına dikkat etmek gerekir.

Kullanılabilecek veritabanı sunucularından bahsetmeden önce ADO hakkında bilgi vermek gerekir. ADO, Windows ortamında bir veritabanı arabirimi olup, veritabanı işlerine çözüm getirmek için Microsoft tarafından geliştirilmiştir. ADO aslında Microsoft SQL Veritabanı Sunucularına özgü bir sürücüdür. Ancak ADO aynı zamanda bir DB2 sürücüsü üzerinden DB2 veritabanına bağlanabileceği gibi, yine ORACLE sürücüsü üzerinden ORACLE veritabanına bağlanabilir. Bundan dolayı istemciye uygulama açısından arka tarafta çalışacak veritabanı sunucusunun bir önemi yoktur.

Ancak veritabanı tarafından bakıldığında bazı önemli mevzular olduğu görülmektedir. Mesela, en basit örnek olarak veritabanında çalışması gereken statik procedürlerin o

veritabanına uygun şekilde yazılması gerekmektedir. Ayrıca tablolarındaki tanımların yine veritabanının desteklediği veritipleri ile oluşturulması şarttır.

Tasarlanan sistem şu anda IBM DB2 7.0 ve Microsoft SQL2000 veritabanı sunucularıyla uyumlu bir şekilde çalışabilmektedir.

Ancak şu belirtilmelidir ki, ADO kullanıldığından dolayı SQL2000 veritabanının DB2'ye göre biraz da olsa bir performans üstünlüğü vardır. Tabii ki DB2'nun da IBM'in büyük makinelerinde çalışmak gibi üstün gücünün olduğunu belirtmek gerekir.

Ancak bu proje daha çok küçük ve orta ölçekli işyerleri için düşünüldüğünden, kütüphane sunucu için en uygun konfigürasyon CPU'su güçlü makinalarda çalışan bir SQL2000 veritabanı sunucusudur.

4.3 Veritabanı Sistem Bilgisinin Oluşturulması

Aşağıda kütüphane sunucusu veritabanında kurulum aşamasında oluşturan ve sistemin katalog bilgisini tutan tabloların yapıları ve ne iş yaptıkları anlatılacaktır.

4.3.1 Veritabanı kolon tipleri

Tablo tanımlarına geçmeden önce bazı veritabanı kolon veri tiplerinin anlatılması tabloların anlaşılmasını kolaylaştıracaktır.

Eğer bir kolonun veritabanı tipi "int" olarak işaretlenmişse bu kolonun tamsayı değerlerinden oluşacağı anlamına gelir. "Varchar(n)" tipli bir kolon 1 ile n karakter uzunluğu arasında kelimelerden oluşacağını ifade eder. Bu veritabanı için dinamik bir yerdir çünkü kelime kaç karakterden oluşuyorsa veritabanında o kadar bir yer ayrılır. "Char(n)" ifadesi "Varchar(n)" ile bir fark hariç tamamen aynıdır. Bu fark ise veritabanının kolon uzunluk değeri ne olursa olsun n karakterlik yer ayracağıdır. Kolon eğer "data" ve "datetime" değerleri sırayla tarih ve zamanı ifade ederler. Tarih gg.aa.yyyy formatında, zaman ise gg.aa.yyyy ss.dd.sn formatındadır.

Eğer bir kolon IDENTITY(n,m) şeklinde ifade edilmişse, kolon otomatik değer alan bir kolondur ve tamsayı n'den başlayıp m tamsayısı kadar arta arta gidecek demektir.

"NOT NULL" ifadesi kolonun boş bırakılamaz olduğunu belirtir.

4.3.2 Giriş kontrol listeleri

Çizelge 4.2 DMSACCESSLISTCODES tablo yapısı

Kolon Adı	Veri Tipi
ACCESS_LIST_ID	Integer IDENTITY(1,1) NOT NULL
ACCESS_LIST_NAME	Varchar(50)

Kütüphane sunucusunun görevlerinden bir tanesi sistemde güvenliği sağlamaktır. Bunun için sisteme hangi kullanıcının hangi indeks sınıfına veya hangi çalışma listesine hangi ayrıcalıklarla gireceğini tuttuğu bazı kontrol listeleri tanımlanması gerekir.

DMSACCESSLISTCODES tablosu kütüphane sunucusunda tanımlı olan giriş kontrol listelerinin tanımlarını tutacak tablodur.

ACCESS_LIST_ID kolonu sistemdeki bir kontrol listesini ifade eden tablosa eşsiz bir tamsayıdır. ACCESS_LIST_NAME ise bu giriş kontrol listesinin adıdır.

4.3.3 Giriş kontrol liste detayları

Bir giriş kontrol listesi içinde bir çok kullanıcı-ayrıcalık çiftleri vardır. Bu çiftlerin tutulması kütüphane sunucu üzerindeki DMSACCESSLISTS denilen tabloda gerçekleşir. Kısaca bu tablo giriş kontrol listelerinin detaylarını tutan tablodur.

Çizelge 4.3 DMSACCESSLISTS tablo yapısı

Kolon Adı	Veri Tipi
ACCESS_LIST_ID	Integer NOT NULL
PATRON_ID	Integer NOT NULL
PATRON_KIND	Smallint NOT NULL
PRIVILEGE_ID	Integer NOT NULL

ACCESS_LIST_ID kolonu hangi giriş kontrol listesinin olduğunu belirler. PATRON_ID kolonu bir sistem kullanıcısının veya bir sistem grubunun kodunu tutar.

PRIVILEGE_ID ise kütüphane sunucusunun DMAPRIVILEGES tablosunda tanımlanan ayrıcalık setlerinden bir tanesinin kodudur.

PATRON_KIND kolonu bu veritabanı kaydının nasıl değerlendireceği bilgisini tutar. 0 değeri belirtilen giriş kontrol listesine tüm kullanıcıların belirtilen ayrıcalıkla gireceğini belirtmek

için kullanılır. Bu durumda PATRON_ID değişkeninin değeri -1 dir ve anlamı yoktur. Bu kolon değeri 1'e eşitlenirse bunun anlamı bu giriş kontrol listesinin PATRON_ID ile belirtilen tanımında gerçekten bir kullanıcı olduğunu ve tanımlı ayrıcalık setinin PATRON_ID'de belirtilen kolonun da kullanıcı için geçerli olduğunu ifade eder. 2 değeri ise giriş kontrol listesinin PATRON_ID ile belirtilen kolonunda bir grup kodunun bulunduğunu ve ayrıcalık setinin bu gruptaki tüm kullanıcılar için uygulanacağını bildirir.

Bu tablo hangi giriş kontrol listelerine hangi kullanıcıların hangi yetkilerle tanımlı olduğu bilgisini tutar. Sistemdeki güvenliğin sağlanabilmesi için en önemli bilgileri taşır. Çünkü indeks sınıfları veya çalışma listeleri tanımlanırken bunlara bir giriş listesi tanımlanmak zorundadır. Giriş listesinin detayları ise doğrudan o indeks sınıfı veya çalışma listesindeki güvenliği belirler.

4.3.4 Anahtar alanları

İndeks sınıfları veritabanı tablolarıdır ve yönetici uygulama kullanılarak bir indeks sınıfı tanımlandığında sistem tarafından kütüphane sunucusunun veritabanında dinamik olarak bir tablo yaratılır.

Çizelge 4.4 DMSATTRIBUTEDEFINITIONS tablo yapısı

Kolon Adı	Veri Tipi
ATTRIBUTE_ID	Integer IDENTITY (1,1) NOT NULL
ATTRIBUTE_NAME	Varchar(50) NOT NULL
SQL_TYPE	Smallint NOT NULL
MINIMUM_SIZE	Smallint NOT NULL
MAXIMUM_SIZE	Smallint NOT NULL

Bu tablonun anlamlı bir hale gelmesi için birden fazla veritabanı alanından oluşması gerekmektedir. Bu veritabanı alanlarından herbirine “anahtar alan” adı verilir. Mesela Musteri_Ad, Musteri_Soyad, Sicil_no gibi. İşte bu veritabanı alanlarının tanımlandığı tablo DMSATTRIBUTEDEFINITIONS adı verilen tablodur.

ATTRIBUTE_ID tabloda eşsiz bir tamsayıdır ve bir veritabanı alanını tamsayı ile ifade etmeye yarar. Bunlardan herbirine sistemde **anahtar alan** adı verilir.

ATTRIBUTE_NAME bu alanın ismidir.

SQL_TYPE bu alanın veri tipini ifade eder. Dört değerden oluşur. 0 karakterler, 1 tamsayı, 2

tarikh, 3 zaman tiplerini ifade eder.

MINIMUM_SIZE ve MAXIMUM_SIZE eğer bu alan karakterler kümesinden oluşuyorsa minimum ve maksimum kaç karakterden oluşmak zorunda olduğunu içerir.

4.3.5 Güncellemenin senkron hale getirilmesi

Birden fazla kullanıcı bir sistemde aynı kaynağa aynı anda birden fazla kullanıcı tarafından güncelleme yapılmasının bir şekilde önlenmesi gerekir.

Bundan dolayı bir kaynak aynı anda güncellenmek için sadece bir kullanıcıya verilmelidir.

Çizelge 4.5 DMSCHECKEDOUT tablo yapısı

Kolon Adı	Veri Tipi
ITEM_ID	Char(32) NOT NULL
PATRON_NAME	Varchar(255) NOT NULL
CREATED	Datetime NOT NULL
EXPIRES	DateTime NOT NULL

Sistemde bu işi gerçekleşmesini sağlayan tablo DMSCHECKEDOUT tablosudur. Bir kullanıcının bir doküman veya bir klasör üzerinde işlem yapabilmesi için dokümanın veya klasörün kullanıcı üzerine check-out (kaynağın başkası tarafından güncellenememe durumu) denilen durumda olması gerekir. Güncelleme işi bitirildikten sonra tutulan nesne başkalarının güncellemesi için serbest bırakılmalıdır (check-in işlemi). Yani check-out bir kullanıcının bu tabloya bir nesneyi yazması, check-in ise silmesi anlamına gelir.

ITEM_ID dokümanın veya klasörün sistemde eşsiz olan 32 karakterlik tanımlayıcı kodudur.

PATRON_ID dokümanın veya klasörün kimin tarafından check-out edildiğini tutan kolondur.

CREATED check-out edilme zamanıdır.

EXPIRES ise dokümanın check-out'unun son bulması gereken maksimum zamanın yazıldığı kolondur. EXPIRES CREATED ile DMSMASTER tablosunun CO_EXPIRE_TIME parametresi ile dakika cinsinde belirtilen değerin toplanması ile hesaplanır. Eğer bir nesne bu zaman aşılmış olduğunda hala serbest bırakılmamışsa sistem bu işi otomatik olarak gerçekleştirir.

4.3.6 İndeks sınıflarının anahtar alanları

DMSCLASSATTRIBUTES tablosu indeks sınıflarının anahtar alanları bilgilerinin tutulduğu tablodur.

Çizelge 4.6 DMSCLASSATTRIBUTES tablo yapısı

Kolon Adı	Veri Tipi
CLASS_ID	Integer NOT NULL
ATTRIBUTE_ID	Integer NOT NULL
SEQUENCE_NUM	Smallint NOT NULL
ATTR_FLAGS	Smallint NOT NULL

CLASS_ID indeks sınıfının kodudur.

ATTRIBUTE_ID sistemde DMSATTRIBUTEDEFINITIONS tablosunda tanımlı olan anahtar alanlardan birinin kodudur.

SEQUENCE_NUM kolonu ise anahtar alanın indeks sınıfının tablosunda kaçınıcı kolon olduğu bilgisini tutar.

ATTR_FLAGS iki değer alır. sıfırın anlamı bu değerın mecburi olduğudur. Bir sayısı ise kolon değerine NULL atanabilir anlamı içerir.

Sistem herhangi bir indeks sınıfın anahtar alan bilgilerine DMSCLASSATTRIBUTES tablosu ile DMSATTRIBUTEDEFINITIONS tablosunu birleştirerek ulaşır.

4.3.7 İndeks sınıfları

İndeks sınıfları doküman veya klasörlerin indeks bilgilerinin tutulduğu tablolardır. Tüm doküman veya klasörler bir indeks sınıfında indekslenmek zorundadırlar. DMSCLASSDEFINITIONS tablosu indeks sınıflarının tanımlarının tutulduğu tablodur.

Çizelge 4.7 DMSCLASSDEFINITIONS tablo yapısı

Kolon Adı	Veri Tipi
CLASS_ID	Integer IDENTITY (1,1) NOT NULL
CLASS_NAME	Varchar(50) NOT NULL
CLASS_TABLE_NAME	Char(8) NOT NULL
ACCESS_LIST_ID	Integer NOT NULL
CREATED	Datetime NOT NULL

CLASS_ID indeks sınıfının tablodaki eşsiz tamsayı kodudur.

CLASS_NAME indeks sınıfının eşsiz adıdır. Ad uzunluğu en fazla 50 karakter olabilir.

CLASS_TABLE_NAME sistem tarafından otomatik olarak üretilecek veritabanı tablo ismidir. Sistemde yeni bir indeks sınıfı yaratıldığında sistem otomatik olarak bir tablo oluşturacaktır. Bu tablonun ismini bu kolonda tutar.

ACCESS_LIST_ID indeks sınıfa tanımlanan giriş kontrol listesinin kodudur.

CREATED indeks sınıfının sistemde yaratılma zamanıdır.

Sisteme giriş yapılırken kullanıcı yetki dahilinde gördüğü indeks sınıf bilgilerini bu tablodan okur.

4.3.8 Doküman tiplerinin belirlenmesi

Sistemin hangi doküman tipleri için destek vereceğinin konfigüre edilmesi gerekir.

Çizelge 4.8 DMSCONTENTCLASSES tablo yapısı

Kolon Adı	Veri Tipi
CONTENT_CLASS	Varchar(8) NOT NULL
LINK_TYPE	Smallint NOT NULL
DO_VERB	Smallint NOT NULL

DMSCONTENTCLASSES doküman yönetim sisteminde hangi doküman tiplerinin saklanabileceğini ve bu doküman tiplerinin standart istemci tarafından nasıl gösterileceğinin tutulduğu tablodur.

CONTENT_CLASS doküman tipidir. TIF, XLS vs.

LINK_TYPE kolonu Windows işletim sistemi üzerinde nesnenin hangi metodla çalıştırılacağını veya gösterileceğini bildirir. Üç ana metod vardır. Bu nesne ya standart istemcinin olanaklarıyla, ya standart istemcinin başka bir programı ayrı bir proses olarak çalıştırmasıyla, yada başka bir programın standart istemcinin içine yüklenmesiyle yönetilir. 0 standart istemci tarafından açılması anlamına gelir. 1 ayrı bir proses olarak başka bir programla açmayı, 2 ise yine başka bir program vasıtasıyla standart istemci içinde açmayı ifade eder . Daha fazla bilgi için Windows OLE nesnelerinin detayları araştırılabilir.

DO_VERB OLE nesnesinin hangi fonksiyonunun çalıştırılması gerektiği bilgisini tutar ve 0

dan başlayarak nesnenin desteklediği fonksiyon sayısına kadar gider.

4.3.9 Sistem kütüğü

Doküman yönetim sistemlerinin en büyük avantajlarından biri yönettiği doküman veya klasörler hakkında detaylı bir tarihçe tutmasıdır. Hangi çalışma listelerine girdikleri veya çıktıkları, üzerlerinde hangi değişikliklerin yapıldığı gibi bilgiler sistemin kaydettiği bilgilerdir.

Çizelge 4.9 DMSHISTORYLOG tablo yapısı

Kolon Adı	Veri Tipi
ITEM_ID	Char(32) NOT NULL
ITEM_ACTION	Varchar(100) NOT NULL
ACTION_TYPE	Smallint NOT NULL
PRIORITY	Integer NOT NULL
TO_WHERE	Varchar(100) NOT NULL
PATRON_NAME	Varchar(255) NOT NULL
ACTION_TIME	Datetime NOT NULL

DMSHISTORYLOG tablosu adındanda anlaşılacağı gibi doküman veya klasörlerin sistem içindeki hareketlerinin tarihçesini tutar.

ITEM_ID'ler doküman veya klasörlerin 32 karakterlik sistemin bütünündeki eşsiz tanımlayıcı kodlarıdır.

ITEM_ACTION değeri sistem tarafından oluşturulan tarihçe ifadesinin tutulması için kullanılır.

ACTION_TYPE hangi tür hareketin yapıldığını gösterir. Mesela bir doküman bir çalışma listesinden çıkartılmış olabilir veya bir doküman bir çalışma listesine yönlendirilmiş olabilir.

PRIORITY dokümanın o anki öncelik değerinin yazıldığı alandır.

TO_WHERE bilgisi nereye yönlendirildiğini tutan bilgidir. Bu bir çalışma listesi adı olabilir.

PATRON_NAME hareketi gerçekleştiren kullanıcının adının tutulduğu kolondur.

ACTION_TIME ise hareketin yapıldığı zamandır.

4.3.10 Nesne not bilgileri

Dokümanlara veya klasörlere not ekleme doküman yönetim sisteminin ihtiyaçlarından birisidir.

Çizelge 4.10 DMSITEMNOTE tablo yapısı

Kolon Adı	Veri Tipi
ITEM_ID	Char(32) NOT NULL
NOTE_LOG	Varchar(3500) NOT NULL

DMSITEMNOTE tablosu bir doküman veya klasörle ilgili bir not saklanabilmesi için oluşturulmuş bir tablodur.

Özellikle doküman bir kullanıcının önünden kaldırılıp diğer bir kullanıcının önüne gelirken yani iş akışında ilerlerken dokümanla özel bazı notların birlikte taşınması gerekebilir. Bu tablo not tutma işi için tasarlanmıştır.

ITEM_ID doküman veya klasör kodu, NOTE_LOG ise yaklaşık 3K lık bir not alanını ifade eder.

4.3.11 Dokümanların fiziksel yer bilgilerinin tutulması

Kütüphane sunucunun temel görevlerinden bir tanesi içerdiği dokümanların doküman sunucular üzerindeki fiziksel yerlerini tutmaktır.

Çizelge 4.11 DMSITEMOBJECTS tablo yapısı

Kolon Adı	Veri Tipi
ITEM_ID	Char(32) NOT NULL
PART	Integer NOT NULL
VERSION	Integer NOT NULL
FILE_SIZE	Integer NOT NULL
CONTENT_CLASS	Varchar(8) NOT NULL
OBJECT_SERVER	Varchar(255) NOT NULL
DISC_NAME	Varchar(20) NOT NULL
PART_NAME	Smallint NOT NULL
CREATED	Datetime NOT NULL
CHANGED	Datetime NOT NULL

DMSITEMOBJECTS tablosu sistemin dokümanların doküman sunucu ile ilgili bilgilerini

tutan tablodur. Dokümanların bu tablo vasıtasıyla nerede oldukları bilinir.

ITEM_ID dokümanın sistemdeki eşsiz kodudur.

PART dokümanın alt parça numarasıdır. VERSION alt parçanın versiyon numarasıdır.

FILE_SIZE dokümanın büyüklüğü, REAL_SIZE dokümanın disk üzerinde kapladığı alandır.

CONTENT_CLASS dokümanın alt parçasının versionunun doküman tipidir.

OBJECT_SERVER dokümanın saklandığı doküman sunucunun adıdır.

DISC_NAME doküman sunucudaki disk adı, PART_NAME ise diskteki mantıksal disk parçasının adıdır.

CREATED dokümanın ilk defa ne zaman oluşturulduğunu, CHANGED ise en son ne zaman güncellendiği bilgisini tutar.

4.3.12 Sistem katalog bilgisi

Sistemdeki tüm doküman veya klasörler bir tabloda indekslenirler. Tablo nesneler için bir katalog bilgisi oluşturur ve özellikle nesnenin hangi indeks sınıfında indekslendiği bilgisi için kullanılır.

Çizelge 4.12 DMSITEMS tablo yapısı

Kolon Adı	Veri Tipi
ITEM_ID	Varchar(32) NOT NULL
ITEM_TYPE	Tinyint NOT NULL
INDEX_CLASS_ID	Integer NOT NULL

ITEM_ID dokümanın veya klasörün 32 karakterli kodu.

ITEM_TYPE nesnenin doküman mı klasör mü olduğunun bilgisini tutar. 0 doküman, 1 klasör anlamındadır.

INDEX_CLASS_ID dokümanın veya klasörün indekslendiği indeks sınıfının kodudur.

4.3.13 Klasör doküman hiyerarşisi

Klasörler ve bunların altındaki diğer klasör veya dokümanların bağlantı hiyerarşisi kütüphane sunucuda tutulması gereken bilgiler arasındadır.

Çizelge 4.13 DMSLINKS tablo yapısı

Kolon Adı	Veri Tipi
PARENT_ID	Varchar(32) NOT NULL
ITEM_ID	Varchar(32) NOT NULL
ITEM_TYPE	Tinyint NOT NULL
INDEX_CLASS_ID	Integer NOT NULL
PARENT_INDEX_CLASS_ID	Integer NOT NULL

DMSLINKS tablosu klasörler ve onlara bağlı diğer klasörler veya dokümanları tutan tablodur.

PARENT_ID 32 karakterlik klasör kodudur.

CHILD_ID 32 karakterlik klasör veya doküman kodudur.

ITEM_TYPE CHILD_ID de bulunan nesnenin klasör veya doküman olduğu bilgisini tutan kolon. 0 doküman, 1 klasör olduğunu bildirir.

INDEX_CLASS_ID ise CHILD_ID nin indekslendiği indeks sınıfının kodudur.

PARENT_INDEX_CLASS_ID PARENT_ID nin indekslendiği indeks sınıfının kodudur.

4.3.14 Sistem parametrelerinin tanımlanması

DMSMASTER sistemdeki bazı önemli parametrelerin tutulduğu tablodur.

Çizelge 4.14 DMSMASTER tablo yapısı

Kolon Adı	Veri Tipi
ACTIVE_OBJ_SERVER	Varchar(255) NOT NULL
ACTIVE_OBJ_DISC	Varchar(20) NOT NULL
ACTIVE_OBJ_PART	Smallint NOT NULL
DEFAULT_FTP_PORT	Integer NOT NULL
DEFAULT_FTP_USER_NAME	Varchar(255) NOT NULL
DEFAULT_FTP_PASSWORD	Varchar(50) NOT NULL
MAX_LOGON	Smallint NOT NULL
MAX_ROW_COUNT	Smallint NOT NULL
CO_EXPIRE_TIME	Integer NOT NULL
LOGON_MESSAGE	Varchar(3500) NOT NULL
LOCK_LOGON	Smallint NOT NULL
LOCK_SYSTEM	Smallint NOT NULL
AVT_COUNT	Smallint NOT NULL

ACTIVE_OBJ_SERVER aktif doküman sunucu adını saklar.

ACTIVE_OBJ_DISC aktif doküman üzerindeki aktif disk adını içerir.

ACTIVE_OBJ_PART aktif doküman üzerindeki mantıksal aktif disk parçasının adını içerir.

DEFAULT_FTP_PORT doküman sunucu üzerinde kurulu olan FTP programının default portunu belirtir.

DEFAULT_FTP_USER_NAME, DEFAULT_FTP_PASSWORD doküman sunucu üzerinde kurulu olan FTP programına giriş yapmak için kullanılan kullanıcı adı ve şifreyi saklar.

MAX_LOGON sisteme giriş için kullanılacak şifrenin maksimum üst üste yanlış girilme sayısıdır. Eğer kullanıcı şifresini MAX_LOGON'dan fazla defa yanlış girerse kullanıcısının sistem kitler ve sistem yöneticisi tarafından şifresinin değiştirilmesi gerekir.

MAX_ROW_COUNT parametresi sistemin performansını ilgilendiren parametrelerden bir tanesidir. İstemci tarafında bir arama yapıldığında veya bir çalışma listesi listelendiğinde eğer bir sınırlama getirilmezse kütüphane sunucusunundan veritabanından istemciye dönen sonuç listesi çok büyük olabilir. Bu sistemin performansını kötü yönde etkileyebilir. Bu parametre ile sunucudan döndürülebilecek maksimum kayıt sayısı sınırlandırılabilir.

CO_EXPIRE_TIME parametresi bir dokümanın maksimum check-out zamanını dakika cinsinden tutar. Bir kullanıcı bir dokümanı veya klasörü check-out ettikten sonra eğer CO_EXPIRE_TIME zamanı içinde check-in etmezse sistem otomatik olarak bu dokümanı veya klasörü check-in ederek serbest bırakır. Bu parametre 60 dakikaya set edilmiştir.

LOGON_MESSAGE sistemin basit parametrelerinden birisidir. Eğer sistem yöneticisi tüm kullanıcılara bir logon mesajı vermek istiyorsa bu parametreye gerekli metni yazar. Kullanıcılar logon olduklarında bu mesajı ekranlarında görürler.

LOCK_LOGON parametresi sisteme yeni bir giriş yapılabilip yapılamayacağını belirler. 0 sisteme girişin serbest olduğunu, 1 ise sisteme girişin yasak olduğunu ifade eder.

LOCK_SYSTEM parametresi sistemi o anda kilitleyip hiçbir güncellemeye izin verilmez. O anda tüm sistem çalışamaz hale gelir. Bu kolon 0'a set edilmeden hiçbir kullanıcı tekrar logon da olamaz. Özellikle bu parametre sistemde acil bir güncelleme yapılması gerekli hallerde sistem yöneticisi tarafından kullanılır. Örnek doküman sunucusu üzerinde yer kalmaması durumudur.

AVT_COUNT sayısı sistem tarafından otomatik oluşturulan dinamik tablolara isim vermek için kullanılan sayıdır. 1000 den başlar ve SOCXXXXX şeklinde sırayla tablo adlarını

oluşturur. SOC01000, SOC01001 gibi.

4.3.15 Doküman sunucu disk parçaları bilgisi

Kütüphane sunucu için dokümanların saklanacağı doküman sunucuları, onların sabit disk ve sabit disk parçaları bilgisini tutmak görevlerinden bir tanesidir.

Çizelge 4.15 DMSOBJECTSERVERPARTITIONS tablo yapısı

Kolon Adı	Veri Tipi
SERVER_NAME	Varchar(255) NOT NULL
DISC_NAME	Varchar(20) NOT NULL
PARTITION_NAME	Smallint NOT NULL
PARTITION_SIZE	Integer NOT NULL
AVAILABLE_SIZE	Integer NOT NULL
PARTITION_STATUS	Char(2) NOT NULL
TRESH_OLD	Integer NOT NULL

DMSOBJECTSERVERPARTITIONS tablosu bir doküman sunucusu üzerindeki mantıksal disk parçaları ve bunların detaylarının tutulduğu tablodur.

SERVER_NAME doküman sunucusunun ismini tutar.

DISC_NAME doküman sunucusunun disk ismini belirler.

PARTITION_NAME ise doküman sunucusunun diskindeki mantıksal disk parçasının ismini belirler. Sistemdeki en küçük doküman saklama birimidir.

PARTITION_SIZE mantıksal disk parçasının MB(MegaByte) cinsinden disk büyüklüğünü tutar.

AVAILABLE_SIZE mantıksal disk parçasının MB cinsinden kullanılmayan alanını ifade eder. Sistem ilk olarak AVAILABLE_SIZE'ı PARTITION_SIZE'a kurar ve daha sonra dokümanlar disk parçasına kaydedildikçe AVAILABLE_SIZE eşik değerine kadar düşer. Bundan sonra sistem SMS sunucuyu kullanarak aktif olarak kullanmak için yeni bir disk parçası seçmek zorundadır.

PARTITION_STATUS disk parçasının durumunu belirtir. Üç satüsü vardır. FU gelecekte kullanılacağını, IU şu anda kullanıldığını, FN kullanılıp bitirildiğini belirtir.

TRESH_OLD değeri, eşik değeri dolayısıyla AVAILABLE_SIZE'ın en küçük değerini

belirtir. Bu değerin altına inildiğinde sistem başka bir disk parçasını aktif olarak işaretlemek zorundadır.

4.3.16 Doküman sunucu disk bilgisi

Çizelge 4.16 DMSOBJECTSERVERDISCS tablo yapısı

Kolon Adı	Veri Tipi
SERVER_NAME	Varchar(255) NOT NULL
DISC_NAME	Varchar(20) NOT NULL
CLUSTER_SIZE	Integer NOT NULL
DISC_STATUS	Char(2) NOT NULL

DMSOBJECTSERVERDISCS tablosu bir doküman sunucusu üzerindeki disk detaylarının tutulduğu tablodur.

SERVER_NAME doküman sunucusunun ismini tutar.

DISC_NAME doküman sunucusunun disk ismini belirler.

DISC_STATUS disk durumunu belirtir. Üç satüsü vardır. FU gelecekte kullanılacağını, IU şu anda kullanıldığını, FN kullanılıp bitirildiğini belirtir.

4.3.17 Doküman sunucu bilgileri

Çizelge 4.17 DMSOBJECTSERVERS tablo yapısı

Kolon Adı	Veri Tipi
SERVER_NAME	Varchar(255) NOT NULL
TYPE	Varchar(10) NOT NULL
PRIORITY	Integer NOT NULL

DMSOBJECTSERVERS tablosu sisteme tanımlı olan doküman sunucularını tutan tablodur.

SERVER_NAME doküman sunucusunun ismini tutar.

TYPE doküman sunucusunun tipini belirler. Kurulu FTP sunucunun tipi için önemlidir. FTP Sunucusu takma ad klasör yapısını destekleyebilir veya desteklemeyebilir. Buna göre istemci tarafı doküman sunucu tarafında nasıl davranacağını belirler.

PRIORITY yeni disk seçiminde hangi doküman sunucunun disk parçalarından birini

kullanılacağını belirleyen öncelik parametresidir.

4.3.18 Kullanıcı grupları

Kullanıcıların yönetiminin kolaylaştırılması amacıyla kütüphane sunucuda kullanıcı grupları oluşturulmuştur.

Çizelge 4.18 DMSPATRONGROUP tablo yapısı

Kolon Adı	Veri Tipi
GROUP_ID	Integer NOT NULL
PATRON_ID	Integer NOT NULL

DMSPATRONGROUP kullanıcı gruplarının detaylarını tutan tablodur. Gruplar içinde başka grupların olmasına izin verilmez. Sadece gruplar ve onlara bağlı kullanıcılar vardır.

GROUP_ID grup kodunu ifade eder. PATRON_ID ise kullanıcı kodunu belirtir.

Gruplar ve kullanıcılar DMSPATRONS grup içinde tutulurlar.

4.3.19 Sistem Kullanıcıları

Kütüphane sunucusu kullanıcıların ve grupların tanımlandığı bir yapı kurmak zorundadır.

Çizelge 4.19 DMSPATRONS tablo yapısı

Kolon Adı	Veri Tipi
PATRON_ID	Integer IDENTITY(1,1) NOT NULL
PATRON_NAME	Varchar(255) NOT NULL
USER_LEVEL	Integer NOT NULL
ADMIN_PRIVILEGES	Char(11) NOT NULL
PASSWORD	Varchar(16) NOT NULL
PASSWORD_EXPIRE_DATE	Varchar(50) NOT NULL
FAILED_LOGONS	Smallint NOT NULL
TYPE_OF_USER	Smallint NOT NULL
LOGGED_ON	Smallint NOT NULL
LOGON_COMPUTER	Varchar(255) NOT NULL
LAST_LOGON	Datetime NULL
LAST_LOGOFF	Datetime NOT NULL

DMSPATRONS sistemdeki kullanıcıları ve grupları tutan tablodur.

PATRON_ID kullanıcı veya grup kodudur.

PATRON_NAME kullanıcı veya grup adıdır.

USER_LEVEL kullanıcılar için geçerli olan kullanıcı seviyesidir. Bu değer doküman güvenliği için kullanılabilir. Bir çalışma listesi listelenirken veya bir arama sonuçları gösterilirken bazı dokümanlar bu değere bakılarak gösterilebilir veya gösterilmeyebilir.

ADMIN_PRIVILEGES kolonu kullanıcının “yönetim uygulaması” üzerinde hangi yetkilerle çalışabileceği bilgisini tutar. 11 karakterden oluşur. 0 paneli açmaz, 1 paneli açar.

- 1.Karakter : Anahtar Alan Panelini
- 2.Karakter : Ayrıcalıklar Paneli
- 3.Karakter : Giriş Kontrol Listeleri Paneli
- 4.Karakter : Kullanıcı Paneli
- 5.Karakter : Gruplar Paneli
- 6.Karakter : İndeks Sınıfları Paneli
- 7.Karakter : Çalışma Listeleri Paneli
- 8.Karakter : Konfigürasyon Paneli
- 9.Karakter : Doküman Sunucu Paneli
- 10.Karakter : SMS Paneli
- 11.Karakter : Admin Hakları Değiştirme Paneli

PASSWORD kullanıcının şifresini kodlanmış bir şekilde saklar.

PASSWORD_EXPIRE_DATE kullanıcı şifresinin geçerliliği olan en son tarihi belirler.

FAILED_LOGONS şifrenin üst üste yanlış yazılma sayını tutar.

TYPE_OF_USER kolonu o kaydın bir kullanıcıya mı veya gruba mı ait olduğunu ifade eder. 0 kullanıcıyı, 1 grubu ifade eder.

LOGGED_ON kullanıcının o anda sistemde çalışıp çalışmadığını tutan kolondur. 0 kullanıcının sistemde olmadığını, 1 ise kullanıcının sistemde olduğunu gösterir.

LOGON_COMPUTER kolonu kullanıcının hangi bilgisayardan giriş yaptığını saklar.

LAST_LOGON en son sisteme giriş tarihini belirtir.

LAST_LOGOFF en son sistemden çıkış tarihini belirtir.

4.3.20 Ayrıcalık setleri

Kütüphane sunucunun temel görevlerinden bir tanesi kontrol listeleri oluşturarak indeks sınıflarının veya çalışma listelerinin güvenliğini maksimum seviyeye getirmektir. Bunun için hak grupları oluşturması gerekir. Bunlara ayrıcalık setleri denir.

Çizelge 4.20 DMSPRIVILEGES tablo yapısı

Kolon Adı	Veri Tipi
PRIVILEGE_ID	Integer IDENTITY(1,1) NOT NULL
PRIVILEGE_NAME	Varchar(50) NOT NULL
PRIVILEGE	Varchar(100) NOT NULL

DMSPRIVILEGES ayrıcalık setlerinin tutulduğu tablodur.

PRIVILEGE_ID ayrıcalık setinin kodunu tutar.

PRIVILEGE_NAME ayrıcalık ismini ifade eder.

PRIVILEGE parametresinin her karakteri sistemdeki doküman veya klasörle ilgili bir ayrıcalığı ifade eder.

Bu ayrıcalıklar Çizelge 4.21’de ayrıntılı olarak verilmiştir.

Çizelge 4.21 Ayrıcalık Detay Açıklamaları

İndeksi	Yetki Adı	Yetki Açıklaması
1	SUPER_ACCESS	Tüm yetkiler
2	DOCUMENT_READ	Doküman Okuma Yetkisi
3	DOCUMENT_UPDATE	Doküman Güncelleme Yetkisi
4	DOCUMENT_CREATE	Doküman Oluşturma Yetkisi
5	DOCUMENT_DELETE	Doküman Silme Yetkisi
6	DOCUMENT_ADD	Doküman Ekleme Yetkisi
7	FOLDER_READ	Klasör Okuma Yetkisi
8	ADD_TO_FOLDER	Klasöre Ekleme Yetkisi
9	REMOVE_FROM_FOLDER	Klasörden Kaldırma Yetkisi
10	FOLDER_CREATE	Klasör Yaratma Yetkisi

11	FOLDER_DELETE	Klasör Silme Yetkisi
12	FORCE_CHECK_IN	Check-In Yetkisi
13	ADD_TO_INDEX	İndeks Sınıfına Ekleme Yetkisi
14	UPDATE_INDEX_INFO	İndeks Bilgisini Güncelleme Yetkisi
15	REMOVE_FROM_INDEX	İndeks Sınıfından Kaldırma Yetkisi
16	SEARCH	İndeks Sınıfında Arama Yetkisi
17	HISTORY_LOG_READ	Doküman Logunu Görüntüleme Yetkisi
18	NOTE_LOG_READ	Doküman Notlarını Okuma Yetkisi
19	NOTE_LOG_UPDATE	Doküman Notlarını Güncelleme Yetkisi
20	WB_READ	Çalışma Listesinde Okuma Yetkisi
21	ADD_TO_WB	Çalışma Listesine Ekleme Yetkisi
22	REMOVE_FROM_WB	Çalışma Listesinden Çıkarma Yetkisi
23	CHANGE_PRIORITY	Öncelik Bilgisini Değiştirme Yetkisi
24	_EXPORT	Dokümanı Sistemden Dışarı Alma Yetkisi
25	_IMPORT	Dokümanı Sisteme Dahil Etme Yetkisi
26	_PRINT	Dokümanı Bastırma Yetkisi

4.3.21 SMS sunucu seçeneklerinin belirlenmesi

Çizelge 4.22 DMSSMSOPTIONS tablo yapısı

Kolon Adı	Veri Tipi
CONTROL_DS	Tinyint NOT NULL
CONTROL_DS_OPTION	Tinyint NOT NULL
CONTROL_DS_VALUE	Tinyint NOT NULL
MAIL_DS	Tinyint NOT NULL
MAIL_DS_OPTION	Tinyint NOT NULL
MAIL_DS_VALUE	Tinyint NOT NULL
CONTROL_OS	Tinyint NOT NULL
CONTROL_OS_OPTION	Tinyint NOT NULL
CONTROL_OS_VALUE	Tinyint NOT NULL
MAIL_WHEN_CONNECT_FAILURE	Tinyint NOT NULL
LOCK_SYSTEM_WHEN_NO_SPACE	Tinyint NOT NULL

MAIL_SERVER	Varchar(255) NOT NULL
USER_ID	Varchar(255) NOT NULL
PORT	Integer NOT NULL
MM_TO	Varchar(500) NOT NULL
MM_CC	Varchar(500) NOT NULL

DMSSMSOPTIONS tablosu sisteminde disk yönetimini sağlayan SMS sunucunun seçeneklerini belirleyen tablodur.

CONTROL_DS SMS sunucunun aktif disk parçasını kontrol edip etmemesi gerektiğini ifade eder. 0'ın anlamı kontrol edilmesinin gerekli olmadığı, 1'in anlamı ise kontrol edilmesi gerektiğidir. CONTROL_DS_OPTION ise eğer CONTROL_DS 1'e kurulmuş durumda ise bu kontrol işleminin ne şekilde yapılacağını belirler. 0'ın anlamı günde tek bir kontrol, 1'in anlamı ise periyodik bir kontroldür. CONTROL_DS_VALUE ise CONTROL_DS_OPTION'a göre anlam kazanır. Eğer CONTROL_DS_OPTION periyodik bir değer gösteriyorsa buradaki değer periyodik olarak bu kadar saatte bir kontrolü ifade eder. Ama CONTROL_DS_OPTION günde bir kez kontrolü işaret ediyorsa bu kolondaki değer bu kontrolün 24 saat üzerinden hangi saatte yapılacağını bildirir.

MAIL_DS SMS sunucunun aktif disk parçasını kontrol ederek disk statülerini maillemesi gerekip gerekmediğini ifade eder. 0'ın anlamı kontrol edilmesinin gerekli olmadığı, 1'in anlamı ise kontrol edilmesi gerektiğidir. MAIL_DS_OPTION ise eğer MAIL_DS 1'e kurulmuş durumda ise bu kontrol işleminin ne şekilde yapılacağını belirler. 0'ın anlamı günde tek bir mail, 1'in anlamı ise periyodik bir maildir. MAIL_DS_VALUE ise MAIL_DS_OPTION parametresinin değerine göre anlam kazanır. Eğer MAIL_DS_OPTION periyodik bir değer gösteriyorsa MAIL_DS_VALUE'deki değer periyodik olarak bu kadar saatte bir disk statülerinin maillenmesini ifade eder. Ama MAIL_DS_OPTION günde bir kez maili işaret ediyorsa bu kolondaki değer bu mailin 24 saat üzerinden hangi saatte yapılacağını bildirir.

CONTROL_OS seçeneği SMS sunucunun doküman sunucuların FTP sunucu durumlarının kontrol edilmesi gerekip gerekmediğini ifade eder. 0'ın anlamı kontrol edilmesinin gerekli olmadığı, 1'in anlamı ise kontrol edilmesi gerektiğidir. CONTROL_OS_OPTION ise eğer CONTROL_OS 1'e kurulmuş durumda ise bu kontrol işleminin ne şekilde yapılacağını belirler. 0'ın anlamı günde tek bir kontrol, 1'in anlamı ise periyodik bir kontroldür. CONTROL_OS_VALUE ise CONTROL_OS_OPTION'a göre anlam kazanır. Eğer CONTROL_DS_OPTION periyodik bir değer gösteriyorsa CONTROL_OS_VALUE parametresindeki değer periyodik olarak bu kadar saatte bir doküman sunucuların kontrolünü

ifade eder. Ama CONTROL_OS_OPTION günde bir kez kontrolü işaret ediyorsa bu kolondaki değer bu kontrolün 24 saat üzerinden hangi saatte yapılacağını bildirir.

MAIL_WHEN_CONNECT_FAILURE parametresi doküman sunuculardan birine bağlantı kontrolü esnasında sorun yaşanmışsa bu durumun maillenmesi gerekip gerekmediğini ifade eder. 0 gerekmediğini, 1 ise maillenmesi gerektiğini bildirir.

LOCK_SYSTEM_WHEN_NO_SPACE parametresi SMS sunucunun kontrolü sırasında sistem doküman sunucuların hiçbirisinde yer kalmadı ise bu durumda tüm sistemi kilitleyip kilitlememesi durumunu ifade eder. 0 set edilirse sistem kilitlenmez. Ancak 1'e kurulursa tüm kullanıcılar sistemden çıkış durumuna getirilerek tüm sistem kilitlenir. Bu parametreyi 1'e kurmakta fayda vardır.

MAIL_SERVER parametresi SMS sunucunun mail atması gereken durumda hangi mail sunucunun kullanılacağını belirten parametredir.

USER_ID parametresi MAIL_SERVER'da belirtilen mail sunucuya hangi kullanıcı ile bağlanılacağını saklar.

PORT parametresi MAIL_SERVER parametresi ile belirtilen e-mail sunucunun hangi portu dinlediğini ifade eder.

MM_TO kolonu mailin hangi mail adreslerine direk olarak gönderileceğini tutan text alandır.

MM_CC kolonu mailin hangi mail adreslerine kopya olarak gönderileceğini tutan text alandır.

4.3.22 Çalışma liste detayları

Kütüphane sunucusunun doküman veya klasörlerin iş akışının yürütülebilmesi amacıyla çalışma listelerini yönetebilecek bir yapıyı sağlaması gerekir.

Çizelge 4.23 DMSWIPITEMS tablo yapısı

Kolon Adı	Veri Tipi
WORKBASKET_ID	Char(32) NOT NULL
ITEM_ID	Char(32) NOT NULL
INDEX_CLASS_ID	Integer NOT NULL
ITEM_TYPE	Tinyint NOT NULL
PRIORITY	Integer NOT NULL
ITEM_INFO	Varchar(4096) NOT NULL
CREATED	Datetime NOT NULL

DMSWIPITEMS tablosu sistemdeki çalışma listelerinin deyatlarının tutulduđu tablodur. Çalışma listeleri içinde hem klasör hem de dokümanlar olabilir.

WORKBASKET_ID 32 karakterden oluşan sistem bazında eşsiz bir koddan oluşturulmuştur.

ITEM_ID 32 karakterden oluşan doküman yada klasör kodudur.

ITEM_CLASS_ID ise ITEM_ID ile belirtilen nesnenin hangi indeks sınıfında indekslendiğini ifade eder.

ITEM_TYPE parametresi ITEM_ID de belirtilen nesnenin bir doküman mı veya bir klasör mü olduğunu ifade eder. 0 doküman,1 klasör anlamında kullanılır.

PRIORITY dokümanın öncelik bilgisini tutar.

ITEM_INFO metin alanı performans için geliştirilmiş bir alandır. Doküman veya klasörün indeks bilgilerini tutan alandır.

CREATED bilgisi ise bu doküman veya klasörün bu çalışma listesine ne zaman eklendiği bilgisini tutar.

4.3.23 Çalışma listeleri

Çizelge 4.24 DMSWORKBASKETS tablo yapısı

Kolon Adı	Veri Tipi
WORKBASKET_ID	Char(32) NOT NULL
WORKBASKET_NAME	Varchar(50) NOT NULL
ACCESS_LIST_ID	Integer NOT NULL
SYSTEM_ASSIGNED	Tinyint NOT NULL
CREATED	Datetime NOT NULL

DMSWORKBASKETS sistemdeki çalışma listelerini tutan tablodur.

WORKBASKET_ID 32 karakterden oluşan sistem bazında eşsiz bir koddan oluşturulmuştur.

WORKBASKET_NAME çalışma listesinin ismidir.

ACCESS_LIST_ID giriş kontrol listesinin kodudur.

SYSTEM_ASSIGNED bu çalışma listesinin çalışma mantığıdır. Eğer çalışma listesi

SYSTEM_ASSIGNED ise çalışma listesi havuz mantığında olacak, listelenmeyecektir. İstemci tarafında çalıştırılacak GetNextWorkbasketItem fonksiyonu ile belirtilen çalışma listesinde önceliği ilk olan eleman kullanıcıya döndürülecektir. Eğer SYSTEM_ASSIGNED değilse, bu çalışma listesini kullanıcılar listeleterek bu listeden istedikleri dokümanı seçme hakkına sahip olacaklardır. Ancak çalışma listesini listelemek, çalışma listesinden bir doküman seçmekten çok daha pahalı bir iştir.

CREATED çalışma listesinin yaratılma zamanını gösterir.

4.4 Veritabanı Dinamik Tabloları

Tasarlanan doküman yönetim sisteminin merkezi olan kütüphane sunucusunda kurulum sırasında yaratılan tabloların yanısıra, yeni indeks sınıfları yaratıldıkça bazı dinamik tablolar da oluşturulur.

Yeni bir indeks sınıfı yaratıldığında sistem DMSMASTER tablosunun AVT_COUNT kolonunda varolan değere bir ekleyerek “SOC” önekinin arkasına 5 karakter şeklinde ekleme yaparak yeni tablo ismini oluşturur ve DMSCLASSDEFINITIONS adlı tabloya yeni indeks sınıfını eklerken CLASS_TABLE_NAME kolonuna da bu SOCXXXXX değerini yazar. Artık bu tablo bu indeks sınıfının içinde bulundurulacak doküman veya klasör bilgilerinin indekslenmesinde kullanılacaktır.

Yeni oluşturulan SOCXXXXX tablosunun ilk kolonu daima ITEM_ID char(32) şeklinde olacaktır ve bu değer veritabanı sunucusuna bir eşsiz indeks olarak tanımlanacaktır. Diğer alanlar ise tamamen sistem yöneticisinin bu indeks sınıfında hangi anahtar alanları görmek istediği ile ilgilidir. Yönetici program kullanılarak eklenen kolonlar birer birer tabloya kolon olarak eklenir.

Dikkat edilmesi gereken eğer bu indeks sınıfının belli kolonları üzerinde arama yapılacağı biliniyorsa o kolonlara veritabanı sunucusunun yönetici programları kullanılarak indeks tanımlanması gerektiğidir. Aksi takdirde her aramada performansı çok kötü bir şekilde etkileyebilen “tüm tablo taramaları” ile karşılaşılabilir.

Bir örnekle detaylandıralım;

DMSMASTER tablosunun AVT_COUNT değişkeninde 1000 değeri olsun. Sistem yöneticisi “BIM_INDEX” adında yeni bir indeks sınıfı açacak olsun. Bu BIM_INDEX içinde Sicil_No(integer),Musteri_Ad(varchar(20)),Musteri_Soyad(varchar(20)),CREATED(datetime

) alanları bulunsun.

Sistem yöneticisi yönetici programda bu tanımları yaptığında kütüphane sunucusunun veritabanında (Microsoft SQL) SOC01000 adında aşağıdaki çizelgedeki bir tablo tanımlanır. Bunu yönetici program otomatik olarak yapar.

Çizelge 4.25 Örnek dinamik tablo yapısı

Kolon Adı	Veri Tipi
ITEM_ID	Char(32)
Sicil_No	Integer
Musteri_Ad	Varchar(20)
Musteri_Soyad	Varchar(20)

Bu indeks sınıfında değişiklik yapmak demek bu kolonlarda silinme veya ekleme yapmak demektir. Bu veritabanı açısından pek doğru bir işlem değildir. Bunun için indeks sınıflarının dikkatli bir şekilde tasarlanmaları gerekir. DB2 veritabanı kullanan sistemlerde veritabanı tablosundan bir kolon silmek mümkün değildir!

4.5 Statik Prosedürler

Bir kütüphane sunucusunun taşınması gereken en önemli özelliklerinden bir tanesi performans parametrelerinin en iyi şekilde tasarlanmış. Bundan dolayı tasarlanan sistemde elden geldiğince işlerin veritabanını tarafında yazılan statik prosedürlerle yapılmasına özen gösterilmiştir.

Aşağıdaki tabloda çalışan statik prosedürler ve bunların ne iş yaptıkları verilmiştir. Bu statik prosedürlerden bazıları yönetici uygulaması tarafından kullanılırken, bir takımını SMS Sunucu, diğer bir takımını ise İstemci tarafından kullanılmaktadır.

Aşağıdaki statik prosedürlerin DB2 da da Microsoft SQL de de adları aynıdır.

4.5.1 Yönetici program kütüphane sunucusu statik prosedürleri

Çizelge 4.26 YP tarafından kullanılan statik prosedürleri

Yönetici Uygulama Tarafınan Kullanılan SP'ler	Detay
SP_ADM_ACL_AVAILABLE_USERS	Bir giriş kontrol listesine

	eklenebilecek kullanıcıları bulur.
SP_ADM_ADD_USER_TO_GROUP	Bir kullanıcıyı bir gruba ekler.
SP_ADM_AVAILABLE_USERS	Bir grupta olmayan kullanıcıları listeler.
SP_ADM_CONTROL_DELETE_DISC	Bir diskin silinmeye uygun olup olmadığını kontrol eder.
SP_ADM_CONTROL_OBJECT_SERVER_DELETE	Bir doküman sunucunun silinmeye uygun olup olmadığını kontrol eder.
SP_ADM_CONTROL_PASSWORD	Admin şifresini kontrol eder.
SP_ADM_DELETE_ACL	Giriş listesini siler.
SP_ADM_DELETE_ACL_DETAIL	Giriş listesi detaylarını siler.
SP_ADM_DELETE_ATTRIBUTE	Verilen anahtar alanı siler.
SP_ADM_DELETE_CLASS_ATTRIBUTE	Bir indeks sınıfın bir anahtar alanını siler.
SP_ADM_DELETE_INDEX_CLASS	İndeks sınıfını siler.
SP_ADM_DELETE_OBJECT_SERVER	Doküman sunucu kaydını siler.
SP_ADM_DELETE_OBJECT_SERVER_DISC	Doküman sunucu disk kaydını siler.
SP_ADM_DELETE_OBJECT_SERVER_PARTITION	Doküman sunucu diskinin disk parçasını siler.
SP_ADM_DELETE_PATRON	Kullanıcı kaydını siler.
SP_ADM_DELETE_PRIVILEGE	Privilege kaydını siler.
SP_ADM_DELETE_USER_GROUPS	Kullanıcıya ait grupları siler.

SP_ADM_DELETE_WORKBASKET	Çalışma listesini siler.
SP_ADM_GET_ALL_FTP_SERVERS	Tüm doküman sunucularını listeler.
SP_ADM_INSERT_ATTRIBUTE	Anahtar alan ekler.
SP_ADM_INSERT_CLASS_ATTRIBUTE	İndeks sınıfa anahtar alan ekler.
SP_ADM_INSERT_INDEX_CLASS	İndeks sınıfı ekler.
SP_ADM_INSERT_NEW_OBJECT_SERVER	Yeni doküman sunucu ekler.
SP_ADM_INSERT_OBJECT_SERVER_DISC	Doküman sunucu diski ekler.
SP_ADM_INSERT_PARTITION	Doküman sunucu disk parçası ekler.
SP_ADM_INSERT_PATRON	Kullanıcı ekler.
SP_ADM_INSERT_PATRON_TO_ACL	Kullanıcıyı giriş listesine ekler.
SP_ADM_INSERT_PRIVILEGE	Ayrıcalık seti ekler.
SP_ADM_INSERT_WORKBASKET	Çalışma listesi ekler.
SP_ADM_REMOVE_PATRON_FROM_GROUP	Kullanıcıyı gruptan çıkarır.
SP_ADM_SELECT_ACLS	Tüm giriş kontrol listelerini listeler.
SP_ADM_SELECT_ACL_PUBLICS	Giriş kontrol listesinin genel tanımlanmış tüm ayrıcalık setlerini listeler.
SP_ADM_SELECT_ACL_USERS	Giriş kontrol listesinin detaylarını seçer.
SP_ADM_SELECT_ALL_GROUPS	Tüm kullanıcı gruplarını seçer.

SP_ADM_SELECT_ALL_OBJECT_SERVERS	Tüm doküman sunucuları listeler.
SP_ADM_SELECT_ALL_PARTITIONS	Tüm doküman sunucuların tüm disk parçalarını listeler.
SP_ADM_SELECT_ATTRIBUTES	Tüm anahtar alanları seçer.
SP_ADM_SELECT_CLASS_ATTRIBUTES	İndeks sınıfının tüm anahtar alanlarını seçer.
SP_ADM_SELECT_GROUP_USERS	Tüm grup kullanıcılarını seçer.
SP_ADM_SELECT_INDEX_CLASSES	Tüm indeks sınıflarını seçer.
SP_ADM_SELECT_MAX_PART_COUNT	Maksimum disk parça numarasını seçer.
SP_ADM_SELECT_OBJECT_SERVER_DISCS	Doküman sunucunun disklerini seçer.
SP_ADM_SELECT_OBJECT_SERVER_PARTITIONS	Doküman sunucunun tüm disk parçalarını seçer.
SP_ADM_SELECT_PATRONS	Tüm kullanıcıları seçer.
SP_ADM_SELECT_PRIVILEGES	Tüm ayrıcalık setlerini seçer.
SP_ADM_SELECT_SMS_OPTIONS	Tüm SMS seçeneklerini listeler.
SP_ADM_SELECT_USER_GROUPS	Bir kullanıcının dahil olduğu grupları bulur.
SP_ADM_SELECT_WORKBASKETS	Çalışma listelerini listeler.
SP_ADM_SET_SERVER_DISC_TRESH_OLD	Server disk parçasının treshold değerini kurar.
SP_ADM_UPDATE_ATTRIBUTE	Anahtar alanı günceller.

SP_ADM_UPDATE_CLASS_ATTRIBUTE	İndeks anahtar alanını günceller.
SP_ADM_UPDATE_CONFIGURATION	Sistem konfigürasyonunu içinde bulunduran DMSMASTER tablosunu günceller.
SP_ADM_UPDATE_DISC_STATUS	Disk durumlarını günceller.
SP_ADM_UPDATE_INDEX_CLASS_ACL	Bir indeks sınıfının giriş kontrol listesini günceller.
SP_ADM_UPDATE_OBJECT_SERVER_PRIORITY	Doküman sunucu öncelik değerini değiştirir.
SP_ADM_UPDATE_PARTITION_STATUS	Doküman sunucunun bir diskinin bir parçasının statüsünü günceller.
SP_ADM_UPDATE_PATRON	Kullanıcıyı günceller.
SP_ADM_UPDATE_PRIVILEGE	Ayrıcalık setini günceller.
SP_ADM_UPDATE_SMS_OPTIONS	SMS seçeneklerini günceller.
SP_ADM_UPDATE_WORKBASKET	Çalışma listesinin değerlerini günceller.

4.5.2 SMS Sunucu kütüphane sunucusu statik prosedürleri

Çizelge 4.27 SMS tarafından kullanılan statik prosedürleri

SP İsmi	Detay
SP_SMS_GET_NEW_PARTITION	Bir disk parçasının yeri tükendiğinde yeni bir disk parçasının seçimini yapan statik

	prosedür.
SP_SMS_GET_PARTITION_SIZE	Bir disk parçasının büyüklüğünü bulur.
SP_SMS_LOCK_SYSTEM	Tüm kullanıcıların logonunu keserek sistemi kilitler.
SP_SMS_SELECT_IU_PARTITION	Kullanılan disk parçasını bulur.
SP_SMS_UPDATE_ACTIVE_PARTITION	Aktif parçayı işaretler.

4.5.3 İstemci kütüphane sunucusu statik prosedürleri

Çizelge 4.28 İstemci tarafından kullanılan statik prosedürler

SP İsmi	Detay
SP_SYS_ADD_PART	Bir dokümana yeni bir alt doküman parçası eklenmesi. DMSITEMOBJECTS tablosu.
SP_SYS_CHANGE_INDEX_CLASS	Bir doküman veya klasörün indeks değiştirir.
SP_SYS_CREATE_PART	Sistemde bir doküman kaydı yaratır.
SP_SYS_DELETE_ALL_PARTS	Bir dokümanın DMSITEMOBJECTS tablosundaki tüm kayıtlarını siler.
SP_SYS_DELETE_PART	dokümanın DMSITEMOBJECTS tablosunda ilgili alt parçası ile ilgili tüm kayıtlarını siler.

SP_SYS_DELETE_PART_VERSION	Bir dokümanın DMSITEMOBJECTS tablosunda ilgili alt parçasının ilgili versiyonu ile ilgili tüm kayıtlarını siler.
SP_SYS_GET_ALL_ATTRIBUTES	Tüm anahtar alanları listeler.
SP_SYS_GET_CHECKED_STATUS	Bir doküman veya klasörün check-out olup olmadığını kontrol eder.
SP_SYS_GET_HISTORY_LOG	Bir doküman veya klasörün hareket logunu listeler.
SP_SYS_GET_INDEX_CLASS_KEY_FIELDS	Bir İndeks sınıfın anahtar alanlarını listeler.
SP_SYS_GET_ITEM	Bir doküman veya klasörün hangi indeks sınıfında olduğunu ve check-out durumunu gösterir.
SP_SYS_GET_ITEM_OBJECT_STORE_INFO	Bir dokümanın hangi doküman sunucu ve bu doküman sunucusunun hangi disk parçasında olduğunu gösterir.
SP_SYS_GET_ITEM_WORKBASKET_NAME	Bir doküman veya klasörün hangi çalışma listesinde olduğunu gösterir.
SP_SYS_GET_NOTE_LOG	Bir doküman veya klasörün notlarını gösterir.
SP_SYS_GET_PART	Bir dokümanın alt doküman parçası ve versiyonunda hangi tip doküman tipinin olduğunu gösterir.

SP_SYS_GET_PARTS_AND_VERSIONS	Bir dokümanın hangi alt doküman parçaları ve bu alt doküman parçalarının hangi versiyonlardan oluştuğu bilgisini gösterir.
SP_SYS_GET_SYSTEM_PARAMETERS	Sistem parametrelerini listeler.
SP_SYS_GET_USER_GROUP_INDEX_CLASS	Kullanıcının içinde bulunduğu grupların indeks sınıflarını listeler.
SP_SYS_GET_USER_GROUP_WORKBASKET	Kullanıcının içinde bulunduğu grupların çalışma listelerini listeler.
SP_SYS_INSERT_ITEMS_TABLE	Bir doküman veya klasörü DMSITEMS katalog tablosuna ekler.
SP_SYS_ITEM_ADD_TO_FOLDER	Bir doküman veya klasörü bir klasöre ekler.
SP_SYS_ITEM_CHECK_IN	Bir doküman yada klasörü DMSCHECKEDOUT tablosundan siler.
SP_SYS_ITEM_CHECK_OUT	Bir doküman veya klasörü DMSCHECKEDOUT tablosuna ekler.
SP_SYS_ITEM_DELETE	Bir doküman veya klasör bilgilerini siler.
SP_SYS_ITEM_REMOVE_FROM_FOLDER	Bir doküman yada bir klasörü bir başka klasörden çıkarır.
SP_SYS_ITEM_REMOVE_FROM_WORKBASKET	Bir doküman yada klasörü çalışma listesinden çıkarır.

SP_SYS_ITEM_ROUTE_TO_WORKBASKET	Bir doküman yada klasörü çalışma listesine ekler.
SP_SYS_LOGGED_ON	Bir kullanıcının logon olup olmadığını kontrol eder.
SP_SYS_LOGOFF	Kullanıcıyı sistemden çıkarır.
SP_SYS_LOGON	Kullanıcının sisteme logon işlerini halleder.
SP_SYS_SELECT_CONTENT_CLASSES	Desteklenecek doküman tiplerini listeler.
SP_SYS_SELECT_OBJECT_SERVER_DISCS	Tüm doküman sunucuların disklerini listeler.
SP_SYS_SET_NOTE_LOG	Bir doküman yada folder için not yazar.
SP_SYS_SET_PASSWORD	Bir kullanıcının şifresinin değiştirmesini sağlar.
SP_SYS_SET_PRIORITY	Bir doküman yada klasörün çalışma listesindeki öncelik bilgisini set eder.
SP_SYS_UPDATE_ITEM_INFO	Bir dokümanın DMSWIPITEMS tablosundaki ITEM_INFO alanını günceller.
SP_SYS_UPDATE_PART_CONTENT	Bir dokümanın doküman tipini günceller.
SP_SYS_UPDATE_PART_VERSION_CONTENT	Bir dokümanın bir doküman parçasının bir versiyonunu günceller.

4.6 Dinamik Olarak Yaratılan Statik Prosedürler

Bir indeks sınıfı yaratıldığında bu indeks sınıfı içindeki doküman veya klasör bilgilerinin daha verimli ve daha hızlı okunabilmesi için seçmek ve silmek olmak üzere iki tane kütüphane sunucusunun veritabanında dinamik olarak yaratılan statik prosedüre ihtiyaç vardır.

Yani kütüphane sunucusunun veritabanında tanımlanan n tane indeks sınıfı için 2*n tane statik prosedür oluşturulur.

Mesela STR01002 veritabanı tablosuna sahip olan BIM_INDEX adlı bir indeks sınıfı oluşturulduğunda yönetici program otomatik olarak;

spDELETE_ITEM_STR01002

ve

spSELECT_ITEM_STR01002

statik procedürelerini aşağıdaki gibi yaratır.

spDELETE_ITEM_STR01002

```
CREATE PROCEDURE spDELETE_ITEM_STR01002
    @ITEMID CHAR(32)
AS
DELETE FROM STR01002
WHERE ITEM_ID = @ITEMID
RETURN @@ERROR
```

spSELECT_ITEM_STR01002

```
CREATE PROCEDURE spSELECT_ITEM_STR01002
    @ITEMID CHAR(32)
AS
SELECT * FROM STR01002
WHERE ITEM_ID = @ITEMID
```

4.6.1 NOINDEX

Tasarlanan sisteminin kütüphane veritabanı kurulum sırasında NOINDEX adlı, dokümanların ilk defa oluşturulması için kullanılacak, bir indeks sınıfı yaratılacaktır.

Bu indeks sınıfının yapısı çizelge 4.1’de verilmiştir.

İstemci tarafında yeni bir doküman yaratılırken CreateDocument fonksiyonu kullanılır. Bu fonksiyonun yaptığı iş bu tabloya bir kayıt eklemektir.

ITEM_ID’yi istemci tarafı kendisi üretecektir.

SOURCE kısmına genellikle bu kayıt hangi kaynak tarafından oluşturulduğu yazılır. Aslında bu alana anlam ifade edebilecek istenilen bir bilgi yazılabilir.

USERID kısmına o anda istemci hangi kullanıcı ile logon olmuşsa o getirilir. TIMESTAMP ise otomatik olarak bu dokümanın yaratıldığı zamandır.

Doküman veya klasör ilk olarak bu indekste yaratıldıktan sonra NT/2000 tarafındaki uygulamaya bağlı olarak bu indeks sınıfında tutulabileceği gibi başka indeks sınıflarına da taşınabilir.

5. DÖKÜMAN SUNUCULARIN MİMARİSİ

Doküman sunucular sistemde dosya sunucu görevini yapan sunuculardır. Ana görevlere doküman saklamak veya istemciye istediği dokümanı yollamaktır.

Tüm mantık istemci bileşenleri üzerindedir. Doküman sunucuda sadece dosya transferi için konfigüre edilmiş bir FTP sunucu kuruludur.

5.1 Doküman Sunucu Üzerindeki FTP Sunucu

Doküman sunucular sistemdeki tüm doküman nesnelerinin tutulduğu sunuculardır. Bir doküman yönetim sisteminde birden fazla doküman sunucu olabilmektedir.

Doküman sunucuların yaptıkları iş en temel düzeyde doküman saklamak veya istenilen dokümanı isteyen istemciye transfer etmektir.

Bunun için her doküman sunucuda FTP protokolünü destekleyen bir FTP Sunucu çalışmak zorundadır.

Sistemde n tane doküman sunucu olduğunda aynı zamanda n tane FTP sunucu var demektir. Bir FTP sunucu ile bağlantı kurabilmek için 3 bilgiye ihtiyaç duyulur.

1. Kullanıcı ismi
2. Şifre
3. Port numarası

Bu üç parametre tüm ftp sunucular için aynıdır ve değerleri kütüphane sunucusunun veritabanındaki DMSMASTER tablosunun DEFAULT_FTP_USER_NAME, DEFAULT_FTP_PORT ve DEFAULT_FTP_PASSWORD kolonlarında saklıdır.

Sisteme giriş yapan bir istemci bu değerleri alarak doküman sunuculara bağlantılarını gerçekleştirebilir.

5.2 Doküman Sunucu Disklerinin Ulaşımına Açılması

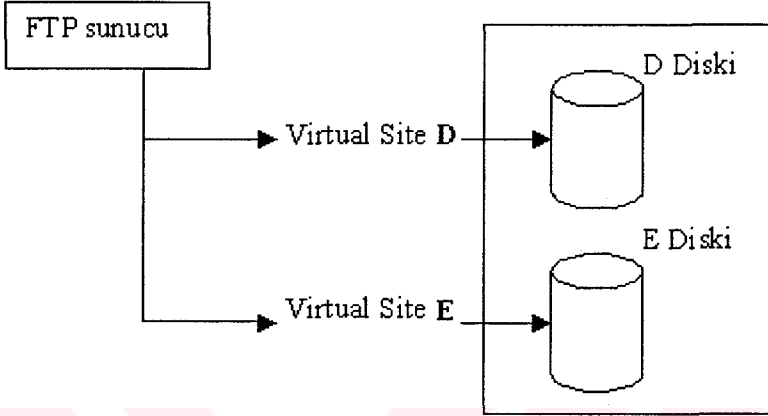
Dokümanların alınması veya verilmesi her doküman sunucu üzerinde kurulması gereken bir FTP sunucu ile gerçekleştirilir.

FTP sunucular iki tipte olabilirler.

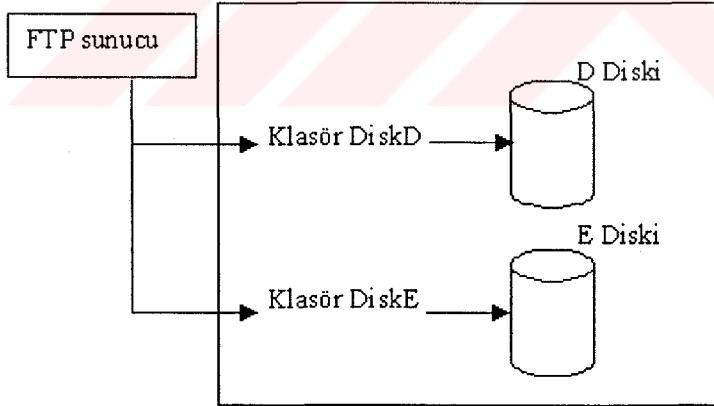
1. FTP sunucunun farklı disk alanlarına takma isimlerle yeni sunucular açabilmesi.

(Örnek Virtual Host destekleyen Microfost FTP Sunucu)

2. FTP sunucunun aynı kök klasörü altındaki farklı klasörlerin işletim sistemi seviyesinde farklı disk alanlarını indeksleyebilmesi. (Unix işletim sistemi)



Şekil 5.1 Takma ad ile farklı disklerin gösterilmesi



Şekil 5.2 İşletim sistemi seviyesinde klasörün disklere linklenmesi gösterilmesi

Şekil 4.1’de görüldüğü gibi FTP sunucu seviyesinde tanımlanan iki farklı takma site ile makina üzerindeki D ve E diski ulaşımı sağlanmıştır.

Bu disk üzerindeki bir dokümana ulaşım gerçekleştirilirken;

FTPKök = FK

Disk İsmi = Takma Site = Dİ

DiskParça İsmi = DPİ

NesneID'nin ilk İki Karakteri = NID2

NesneID = NID

Doküman Alt Parça İsmi = DAPİ

Doküman Alt Parça Versiyon İsmi = DAPVİ

Doküman Tipi = DT

Kısaltmalarını kullanırsak;

Doküman yolu;

FK/TK/Dİ/DPİ/NID2/NID + '.' + DAPİ+ '.' + DAPVİ +DT (3.1)

olur.

Mesela;

Nesne ID = 0E295A4C77A44FA096F3DF5546ABEF34

Disk D

Disk Parç Numarası 1

Part 0

Versiyon 0

Bir TIF dosya

için FTP Kök'lü bir FTP sunucuda doküman yolu

FTP Kök/D/1/0E/0E295A4C77A44FA096F3DF5546ABEF34.0.0.TIF

Olur. Bu yol dokümanın doküman sunucu üzerindeki yerini verecektir.

Bu aşağı yukarı diğer tip ftp sunucu içinde geçerli olacaktır.

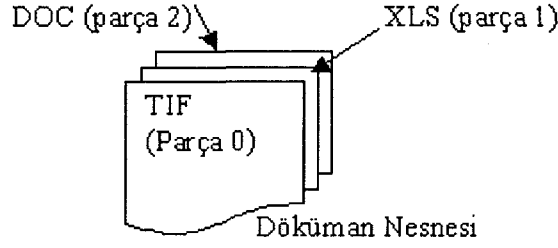
FTP Kök/DiskD/1/0E/0E295A4C77A44FA096F3DF5546ABEF34.0.0.TIF

Doküman istemci tarafında bu yol üzerinden saklanacak veya alınacaktır.

5.3 Doküman Alt Parçaları

Şimdiye kadar doküman sunucuda saklanan her nesneye doküman denildi. Şimdi bu doküman nesnesini biraz daha detaylı açıklayalım.

Doküman nesnesi aslında mantıksal olarak farklı doküman tiplerinden oluşturulabilir. Mesela dokümanın 0. parçasında TIF, 1. parçasında XLS ve 2.parçasında DOC dosyalarını tutuyorum desem ne anlarsınız?



Şekil 5.3 Doküman alt parçalarının oluşturulması

Bunu DMSITEMOBJECTS tablosunu çizerek daha kolay anlatabiliriz.

Çizelge 5.1 Örnek item nesne tablosu (DMSITEMOBJECTS)

ITEM_ID	PART	VERSION	DÖKÜMAN_SUNUCU	DISK_İSMİ	PARÇA	TİP
XXXXXX	0	0	OBJSRV	D	1	TIF
XXXXXX	1	0	OBJSRV	D	1	XLS
XXXXXX	2	0	OBJSRV	D	2	DOC

Yukarıda görüldüğü gibi elimizde bir XXXXX doküman nesnesi olsun. Bu doküman nesnesinin tabloda görüldüğü gibi 0. doküman parçasında bir TIF dosya, 1.doküman parçasında bir XLS dosya ve 2.parçasında bir DOC dosya saklanmıştır.

Yani doküman sunucunun FTP sunucusunun kök klasörü KOK ise,

KOK/D/1/XX/XXXXXX.0.0.TIF

KOK/D/1/XX/XXXXXX.1.0.XLS

KOK/D/2/XX/XXXXXX.2.0.DOC

dosyaları mevcut olacaktır. Bu bir doküman nesnesi içinde birden fazla doküman

tutulabileceği anlamına gelir.

Bu isimlendirme standartını biraz açalım...

FTP Sunucusunun kök klasöründe sonra 'disk adı' gelmektedir. Bundan sonra ise disk parçacığının adı gelir. Bu klasörler alt alta geçildiğinde disk parçacığına ulaşılmış olur. Disk parçacığının altında doküman nesnelerinin kodlarının ilk 2 harfinde oluşturulan klasörler bulunur. Mesela ABF2C4... gibi giden bir doküman koduna sahip doküman AB adlı klasörün altında saklanır. Klasörler tek tek aşıldıktan sonra iş dosyayı bulmaya gelir. Dosya adıda bir standartta göre oluşturulmuştur.

Dosya Adı = NesneKodu + Doküman Parça Numarası + Versiyon Numarası + doküman tipi

Parametrelerinin yanyana gelmesinde oluşturulmuştur.

Böylece doküman sunucudan bir dosya bu standartlara uyarak getirilir veya doküman sunucuya bir dosya bu standartlara uyarak saklanır.

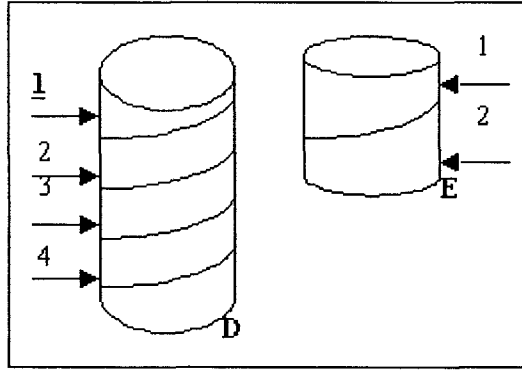
5.4 Mantıksal Disk Parçalarının Oluşturulması ve Kullanılması

Doküman yönetim sisteminde dokümanların saklanması için kullanılan en küçük birime disk parçası adı verilir. Bu tamamiyle mantıksal bir kavramdır.

Disk parçası bir diskin mantıksal olarak n parçaya ayrılmasından oluşur. Her parçaya disk parçası adı verilir. Mesela 90 GB'lık bir kapasiteye sahip olan bir diskin mantıksal olarak 3 parçaya bölünmesi sistem tarafından 30 GB'lık 3 disk olarak bilinmesidir.

Yönetici programını kullanarak sisteme ilk önce bir doküman sunucu tanıtılır. Bu aşamada DMSOBJECTSERVERS tablosuna ekleme yapılır. Daha sonra bu doküman sunucunun doküman saklama amacıyla kullanılacak diskleri tek tek sisteme tanımlanır. Bu tanımlamalarda DMSOBJECTSERVERDISCS tablosuna eklemelere sebep olur. Bundan sonra tanımlana diskler sistem yöneticisinin isteğine göre sistem bu diskleri istenilen parçalara bölerek bu mantıksal parçaları DMSOBJECTSERVERPARTITIONS tablosuna yazar.

Bir örnekle açıklayalım. OBJSRV adlı üzerinde FTP sunucu kurulu ve çalışan bir dosya sunucumuz olsun. Bunun 2 tane D ve E adında diskleri bulunsun. Bu disklerin kapasiteleri sırayla 60GB ve 30GB olsun. İlk diski 4 mantıksal disk parçasına, 2 diski ise 2 mantıksal disk parçasına ayıralım. Böylece elimizde 20GB lık 6 disk parçası olacaktır.



Şekil 5.4 Doküman sunucu disk parçaları

Sistemin SMS sunucusu ilk olarak D diskinin 1 numaralı parçasını aktif disk olarak işaretleyecek ve doküman saklanmak için burası kullanılacaktır. Buranın büyüklüğü 20GB tır.

Kütüphane sunucusunda doküman sunucu ile ilgili üç tablo aşağıdaki şekilde olacaktır.

Çizelge 5.2 Örnek DMSOBJECTSERVERS tablosu

Sunucu	FTP Tipi	Öncelik
OBJSRV	Virtual	1

Doküman sunucu OBJSRV takma adı destekleyen bir FTP sunucuya sahiptir.

Çizelge 5.3 Örnek DMSOBJECTSERVERDISCS tablosu

Sunucu	Disk İsmi	Cluster Byk.	Durum
OBJSRV	D	4096	IU
OBJSRV	E	4096	FU

Koyu işaretli disk artık kullanılmaya başlanmış ve statüsü IU “Kullanılıyor” haline getirilmiştir. Artık bu diski sistemden silmenin bir şansı yoktur.

Buradaki cluster büyüklüğü istemci için çok önemli bir parametredir. Bu parametre vasıtasıyla saklayacağı bir dosyanın diskte kapladığı gerçek yeri hesaplayarak DMSITEMOBJECTS tablosundaki REAL_SIZE değerine yazar. SMS sunucuda bu değer sayesinde bir disk parçasının ne kadarının kullanıldığını ne kadar yer kaldığını hesaplayarak disk parçasının değiştirilip değiştirilmeyeceğine karar verir.

Çizelge 5.4 Örnek DMSOBJECTSERVERPARTITIONS tablosu

Sunucu	Disk İsmi	Parça İsmi	Durum	Byk.	Kalan	Eşik
OBJSRV	D	2	FN	20000	80	100
OBJSRV	D	2	IU	20000	20000	100
OBJSRV	D	3	FU	20000	20000	100
OBJSRV	D	4	FU	20000	20000	200
OBJSRV	E	5	FU	20000	20000	100
OBJSRV	E	6	FU	20000	20000	100

Yukarıdaki Çizelge 5.4’de ise doküman sunucudaki disk parçalarının detayları verilmektedir. Görüldüğü gibi sistem SMS sunucu sayesinde OBJSRV üzerindeki D diskinin 2.disk parçasını IU ‘kullanılıyor’ işaretlemiş ve aktif disk parçası olarak DMSMASTER tablosunda ACTIVE_OBJECT_SERVER, ACTIVE_DISC ve ACTIVE_PART alanlarına yazmıştır.Sisteme giriş yapan bir istemci bu değerleri okuyarak aktif disk parçasının kim olduğunu anlar ve dokümanları oraya saklamaya başlar. OBJSRV’nin 1.disk parçası kullanılmış ve bitirilmiştir. Durumu ‘FN’ olmuştur ve artık kullanılmayacaktır.

Bu 2.disk parçasında **Kalan** kolonu yavaş yavaş azalır ve belli bir süre sonra eşik değerinin altına gelir. O anda SMS sunucu yeni aktif parça olarak OBJSRV sunucusunun D diskinin 3. parçasını aktif disk parçası olarak seçer.

Bu durumda işler biraz karmaşık hal alır. Bazı dokümanlar OBJSRV nin D diskinin 2.parçasında bazı dokümanlar OBJSRV nin D diskinin 3. parçasında bulunmaya başlarlar. 3.disk parçası aktif işaretlendikten sonra asla ve asla sistem 2.disk parçasını doküman saklamak için kullanmaz. Aslında bundan sonra 2.disk parçası tamamiyle sabit ve katı bir yapı halini alır. Çünkü 2. parça bittiğinde sistem operasyonel grubu tarafından bu 2 numaralı disk parçasığının yedeklemesi yapılmış ve kartuş üzerine isim yazılarak kaldırılmış olacaktır. Bunda dolayı eğer 2. disk parçası üzerindeki bir doküman sadece okunmak için alınmış ise doküman buradan alınır. Ancak bir kaydetme işlemi sırasında yeni doküman o andaki aktif disk parçasığına kaydedilir. Burada aynı FTP bağlantısı üzerinden çalışıldığına dikkat ediniz.

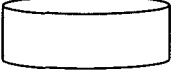
Örneklemeye çalışalım;

D diski üzerinde 2.disk alanı doldu. 3.disk alını aktif olarak işaretlendi.

2.parça dolu



3.parça aktif olarak çalışıyor.



DMSITEMOBJECTS tablosu aşağıdaki gibi olsun;

Çizelge 5.5 Örnek DMSITEMOBJECTS tablosu

ITEM_ID	PART	VERSION	DÖKÜMAN_SUNUCU	DİSK_İSMİ	PARÇA
XXXXX	0	0	OBJSRV	D	2
YYYYY	0	0	OBJSRV	D	2
ZZZZZ	0	0	OBJSRV	D	3
WWWWW	0	0	OBJSRV	D	3

Yukarıdaki örnekte bir istemci XXXXX dokümanını güncellemek istesin. İstemci DMSITEMOCJECTS tablosuna baktığında dokümanı OBJSRV'nin D diskinin 2 numaralı parçasında saklandığını bilir. Ancak sisteme giriş esnasında aldığı değerlerde aktif parça aynı sunucu üzerinde farklı bir disk parçasıdır. Bundan dolayı dokümanı alırken FTP sunucuya bağlandığı noktaya KOK dersek

KOK/D/2/XX/XXXXX yolunu kullanır. Ancak güncelleme yaparsa dokümanı KOK/D/3/XX/XXXXX adıyla kaydeder ve DMSITEMOBJECTS tablosunda XXXXX dokümanının yerini 3 numaralı disk parçası olarak günceller. Böylece 2 numaralı dolu parçaya dokunulmamış olur.

Eğer ZZZZZ kodlu dokümanı alacaksa hem almak hemde saklamak için KOK/D/3/ZZZZZ adlı yolu kullanır. Böylece dosyanın üzerine yazılmış olur.

Sistemi bu denli küçük disklerle bölmenin esas amacı günlük yedekleme işinin minimum zamanda bitirilmesidir. Günlük periyodda sadece aktif parçanın yedeklemesinin yapılması yeterli olacaktır.

Yukarıda aynı doküman sunucunun farklı parçaları üzerinde aynı FTP bağlantı üzerinden giderek dokümanlara ulaşabildik. Peki güncellenmek istenen doküman farklı bir doküman sunucusu üzerinde ve bitirilmiş bir disk parçasında ise ne olacak?

Bununda cevabı yukarıdakine göre fazla karmaşık değildir. İstemci aynı FTP bağlantıyı kullanamayacağından eski dokümanın olduğu doküman sunucunun FTP sunucusuna geçici

bir bağlantı kurarak dokümana ulaşır. Doküman güncellenmek istenirse her zamanki gibi aktif parçaya saklanmak zorundadır. Böylece yine eski disk parçasına başka sunucu üzerinde olsa dokunulmamış olacaktır. Ancak saklanan dokümanın doküman sunucusu kimseye hissettirilmeden değiştirilmiş olacaktır.

5.5 Versiyonlama

Versiyonlama doküman yönetim sisteminde olması gereken özelliklerinden bir tanesidir.

Bilindiği gibi doküman nesneleri doküman alt parçalarından oluşur. Doküman alt parçalarında 0'dan 4 milyara kadar versiyonlanabilir.

Bir nesneye ait doküman parçaları ayrı disk parçalarında veya ayrı disklerde hatta ayrı doküman sunucularda olabildiği gibi doküman alt parçalarına ait versiyonlar bu tür dağıtık bir yapıda olabilir.

Ancak tüm bunların yerleri DMSITEMOBJECTS tablosunda bellidir.

ITEM_ID	PART	VERSION	DÖKÜMAN_SUNUCU	DISK_İSMİ	PARÇA	CLASS
XXXXX	0	0	OBJSRV	D	1	TIF
XXXXX	1	0	OBJSRV	D	1	XLS
XXXXX	1	1	OBJSRV	D	2	XLS
XXXXX	2	0	OBJSRV	D	2	DOC
XXXXX	2	1	OBJSRV2	D	1	DOC
XXXXX	2	4	OBJSRV2	E	2	DOC

Yukarıda XXXXX kodlu doküman nesnesinin farklı doküman parçaları ve bu doküman parçaların farklı versiyonları saklanabilir. Bunlar birbirinden tamamen bağımsız doküman sunucularda, disklerde veya disk parçalarında saklanabilirler. Sistem bu işi arka planda hiç kimseye hissettirmeden yapacaktır.

5.6 Yedekleme

Doküman sunucuların disk parçaları seviyesinde bölünmesinin esas amacı yedekleme zamanının azaltılması ve kolaylaştırılmasıdır.

Disk parçalarının maksimum büyüklüğü yedeklemeyi yapacak kartuş yazıcının hızınada bağlı olmakla beraber 20GB-30GB'lık büyüklüklerde tutulmasında fayda vardır.

Sistem aynı anda sadece bir disk parçasını aktif tutar. Bu aynı anda sadece bir doküman sunucu üzerindeki bir disk parçasının güncelleme için kullanılıyor olması anlamına gelir. Yani hergün yapılacak yedeklemede sadece bir doküman sunucu üzerindeki bir disk parçasının yedeklemesinin yapılması yeterli olacaktır.

Sistemin modüllerinden bir tanesi olan ve sistemin disk alan yönetimini yapan SMS sunucu yaptığı kontrollerde aktif disk parçası üzerinde yeterli alan görmemesi durumunda bu disk parçasını bitmiş olarak işaretleyip yeni bir disk parçası seçimi yapar. Bu seçim sonucunda bulunduğu diski aktif olarak ilan edecektir. Artık eski disk parçasına bir güncelleme yapılmayacaktır. Yani ne ekleme ne silinme olmayacaktır. Bundan dolayı bu disk parçası son defa karta yedeklendikten sonra bir daha yedeklenmesi gerekmeyecektir. Artık günlük yedeklemede seçilen yeni parçanın yedeklenmesi gerekmektedir. Bu süreç böyle sürüp gider.

Sistemin mantığının yüklü olduğu istemci bileşenleri arka tarafta dokümanlar üzerindeki güncellemeleri nasıl yapacaklarını yönetebilirler. Bu yönetimden tamamiyle onlar sorumludurlar.

5.7 SMS Sunucu

SMS Sunucu dokümanların kaydedileceği doküman sunucular üzerindeki disk alanlarının yönetimini yapan uygulamadır.

SMS Sunucu tek başına sürekli çalışan bir programdır. Açılış esnasında (SMSServer.ini) konfigürasyon dosyasından hangi sunucuya bağlanacağını okuduktan sonra tablo dosyasındanda bu sunucu ile detayları öğrenir ve bağlantısını kurup sunucunun veritabanındaki DMSSMSOPTIONS tablosundaki ilk satırı okur.

Çizelge 5.6 SMS sunucu seçenek tablosu

Seçenek	Anlam
CONTROL_DS	0 : Kontrol yok. 1 :Aktif disk parçası durumunu kontrol et.
CONTROL_DS_OPTION	0 : Aktif disk parça kontrol günde bir kez 1 : Periyodik olarak aktif disk parça kontrol

CONTROL_DS_VALUE	Eğer CONTROL_DS_OPTION 0 sa Hangi saatte kontrol edileceği, Eğer CONTROL_DS_OPTION 1 ise Kaç saatte bir kontrol edileceği.
MAIL_DS	0 : Mail Yok 1: Aktif disk parça durumunu maille
MAIL_DS_OPTION	0 : Mail günde bir kez 1 : Mail Periyodik olarak
MAIL_DS_VALUE	Eğer MAIL_DS_OPTION 0 sa Hangi saatte mailleyeceği, Eğer MAIL_DS_OPTION 1 ise Kaç saatte bir mailleyeceğini ifade eder.
CONTROL_OS	0 : Kontrol Yok 1 : Doküman Sunucuların FTP bağlantılarını kontrol et.
CONTROL_OS_OPTION	0 : FTP bağlantı kontrol günde bir kez 1 : Periyodik olarak FTP bağlantı kontrol
CONTROL_OS_VALUE	Eğer CONTROL_DS_OPTION 0 sa Hangi saatte FTP bağlantılarını kontrol edileceği, Eğer CONTROL_DS_OPTION 1 ise Kaç saatte bir FTP bağlantılarını kontrol edileceğini ifade eder.
MAIL_WHEN_CONNECT_FAILURE	0 : FTP bağlantı başarısızlığında mail atma.

	1 : Eğer FTP bağlantılardan birisinde başarısızlık olurda bunu maille.
LOCK_SYSTEM_WHEN_NO_SPACE	0 : Yer kalmaması durumunda hiç birşey yapma. 1 : Yer kalmadıysa sistemi kilitle
MAIL_SERVER	Hangi mail sunucuyu kullanacak.
USER_ID	Mail sunucuya hangi kullanıcıyla girecek
PORT	Mail sunucunun portu
MM_TO	Mail direk kimlere gidecek
MM_CC	Mail CC olarak kimlere gidecek.

Bu seçeneklerle açılan SMS sunucu yapması gereken işleri yapar. Eğer sistemde yer kalmadıysa DMSOBJECTSERVERS, DMSOBJECTSERVERDISCS ve DMSOBJECTSERVERPARTITIONS tablolarını kullanarak bir yeni disk parçası seçer. Bu seçme işlemi direk olarak DMSOBJECTSERVERS tablosunun öncelik kolonu ile ilişkilidir.

Seçilen aktif disk parçası detayları DMSMASTER tablosunda istemcilerin sisteme giriş esnasında okuyacakları yer olan ACTIVE_OBJ_SERVER, ACTIVE_OBJ_DISC ve ACTIVE_OBJ_PARTITION kolonlarına yazılır.

6. STANDART İSTEMCİ

Tasarlanan sistemde tüm doküman yönetim sistem işlemlerinin gerçekleştirilebilmesi için Windows işletim sistemi üzerinde çalışabilecek bir uygulamaya ihtiyaç vardır. Bu uygulama standart istemci olarak adlandırılır.

5. bölümde doküman yönetim sistemleri ile ilgili fonksiyonlarından bahsettiğimiz standart istemcinin bu bölümde önyüz panellerinden bahsederek bazı önemli görevleri nasıl yerine getirildiğini anlatmaya çalışacağız.

Standart istemcinin başlıca görevleri birden fazla kullanıcıyı aynı doküman yönetim sistemi üzerinde çalıştırabilmek, bu kullanıcılara doküman yönetim sistemlerindeki tüm fonksiyonları kullanma fırsatını verebilmektir.

Buna ek olarak, yeni doküman veya klasörler yaratabilmek, klasörlere diğer klasörleri veya dokümanları ekleyebilmek, bunları indeks sınıflarında indeksleyebilmek, çalışma listelerine yönlendirebilmek veya çalışma listelerinden çıkartabilmek, yeni folderlara ekleyebilmek veya çıkartabilmek sayılabilecek diğer görevleri arasındadır.

Ayrıca doküman veya klasörleri arayıp bulabilmek bir diğer önemli sorumluluğudur. Bir veya birden fazla indeks sınıfı üzerinde arama yapılabilir.

Tüm doküman tipleri için hizmet verebilir. Bunlardan imaj dosyalarını kendi bünyesinde yönetebilir. Ses dosyaları, görüntü dosyaları, word dokümanları, excel dokümanları gibi diğer dosya tipleri, Windows'ta bu tip dosyaları açabilecek programlar yüklü olmak kaydıyla, standart istemci tarafından yönetilebilir. Standart istemci bu dosya tiplerinden bazılarını kendi içinde açıp çalıştırabilir (word, excel dosyaları gibi...). Buna embedding denir. Bazılarını ise o dosyayı çalıştıracak programı ayrı bir proses açarak yönetir (görüntü, ses dosyaları gibi...). Buna linking adı verilir.

Aynı zamanda tüm sistemin güvenliğini de sağlar. Sisteme giriş sırasında indeks sınıflarını ve çalışma listelerini giriş kontrol listelerinde kullanıcı için tanımlanan güvenlik parametreleriyle hafızaya alır. Kullanıcı bu indeks sınıflarında veya çalışma listelerinde bir hareket yapacaksa bu hareket güvenlik parametreleri ile asla çakışmamalıdır. Aksi takdirde sistem bu harekete izin vermez.

Standart istemci çok detaylı ve kullanıcı açısından karmaşık olduğundan genellikle firmalar bu OLE nesnesini kullanarak kendi standartlarında bir istemci uygulaması geliştirirler. Sistemin kendi standart istemci genellikle sistem yöneticileri tarafından kullanılır.

Standart istemci bir MDI uygulamadır. Yani bir Microsoft Word programı gibi ana bir form ve bu form içinde açılan bir çok alt formdan oluşmuştur.

Standart istemci Delphi 5 ile yazılmış bir Windows OLE nesnesi olduğundan tüm uygulama diğer geliştirici araçlar (Visual Basic, Microsoft C, Delphi, Borland C vs...) tarafından kullanılabilir.

6.1 Araç Çubukları

Standart istemci uygulaması üzerinde birden fazla formu, bunlarla beraber birden fazla araç çubuğunu bulundurur. Ancak bunlar genellikle aynı fonksiyonlara hizmet ederler.

Şekil 6.1 de genel olarak doküman yönetim sistemi ile ilgili fonksiyonların yürütüldüğü düğmeler görülmektedir.



Şekil 6.1 Araç çubukları - 1

Bunlar soldan sağa doğru şu işlemi yaparlar.

Birinci düğme sisteme girişi sağlar ve uygulama ilk açıldığında aktif haldedir. Sisteme giriş yapıldıktan sonra pasif hale gelir.

İkinci düğme çalışma listelerini listeler. Üçüncü düğme ise kullanıcının hangi form üzerinde olduğuna göre değişir. Eğer hiçbir form açık değil sadece ana form aktif ise sistemden çıkışı, eğer bir alt form aktif ise o formu kapatmayı sağlar.

Bundan sonraki düğmeler dokümanlar ve/veya klasörlerle ilgili düğme takımlarıdır. Dördüncü düğme sistemde bir doküman yaratma, beşinci düğme bir klasör yaratmak için kullanılır.

Altıncı düğme arama yapma düğmesidir.

Yedinci ve sekizinci düğme sırayla doküman veya klasörü check-in ve check-out yaparlar.

Dokuzuncu düğme bir doküman veya klasörü indeksleme veya indeks bilgilerini değiştirmeye yarar.

Onuncu düğme bir doküman veya klasörü başka bir klasöre ekler.

On birinci düğme bir doküman veya klasörü bir klasör yaratarak yaratılan klasöre ekler.

BİRİNCİ DÜĞME

On ikinci düğme bir doküman veya klasörü bir çalışma listesine yönlendirir.

On üçüncü düğme bir doküman veya klasörü bir klasörden çıkarır.

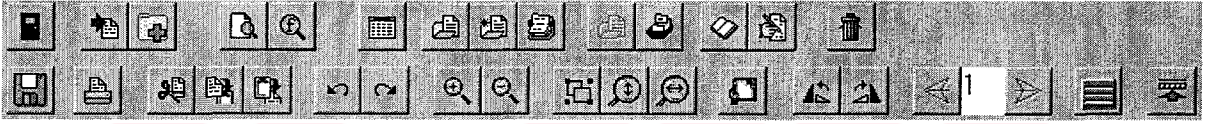
On dördüncü düğme bir klasör veya dokümanı bir çalışma listesinde çıkarır.

On beşinci düğme bir klasör veya dokümanı siler.

On altıncı düğme dokümana veya klasöre eklenmiş notları gösterir.

On yedinci ve son düğme bir doküman veya klasörün tarihçesini gösterir.

Diğer bir tuş takımı ise imaj gösteren form üzerinde bulunur. Bu tuş takımları imaj üzerinde



Şekil 6.2 Araç çubukları – 2

görüntülemeye veya güncellemede kullanılırlar.

Şekil 6.2 de gösterilen tuşlar soldan sağa aşağıda anlatılan işlemleri yaparlar,

Birinci sıra tuşları açılan doküman için Şekil 6.1 için anlatılan işlemlerle aynı işleri yapar.

İkinci sıra takımında;

Birinci düğme dokümanın kaydedilmesini,

İkinci düğme dokümanın yazdırılmasını,

Üçüncü düğme doküman üzerinde bir alanın kesilmesini,

Dördüncü düğme doküman üzerinde bir alanın kopyalanmasını,

Beşinci düğme doküman üzerinde bir alana kesilen veya kopyalanan imajın yapıştırılmasını,

Altıncı düğme dokümanın üzerindeki son değişikliğin geri alınmasını,

Yedinci düğme dokümanın üzerine yapılan son değişikliğin tekrarlanmasını,

Sekizinci düğme görüntüye yaklaşılmamasını,

Dokuzuncu düğme görüntüden uzaklaşılmasını,

Onuncu düğme görüntünün en iyi şekilde yerleştirilmesini,

On birinci düğme görüntünün boya göre yerleştirilmesini,

On ikinci düğme görüntünün ene göre yerleştirilmesini,

On üçüncü düğme görüntü doküman üzerine yazılan notların seçilebilmesini,

On dördüncü düğme görüntünün doksan derece sola döndürülmesini,

On beşinci düğme görüntünün doksan derece sağa döndürülmesini,

On altıncı düğme görüntü sayfa sayfa görüntülenebilmesini,

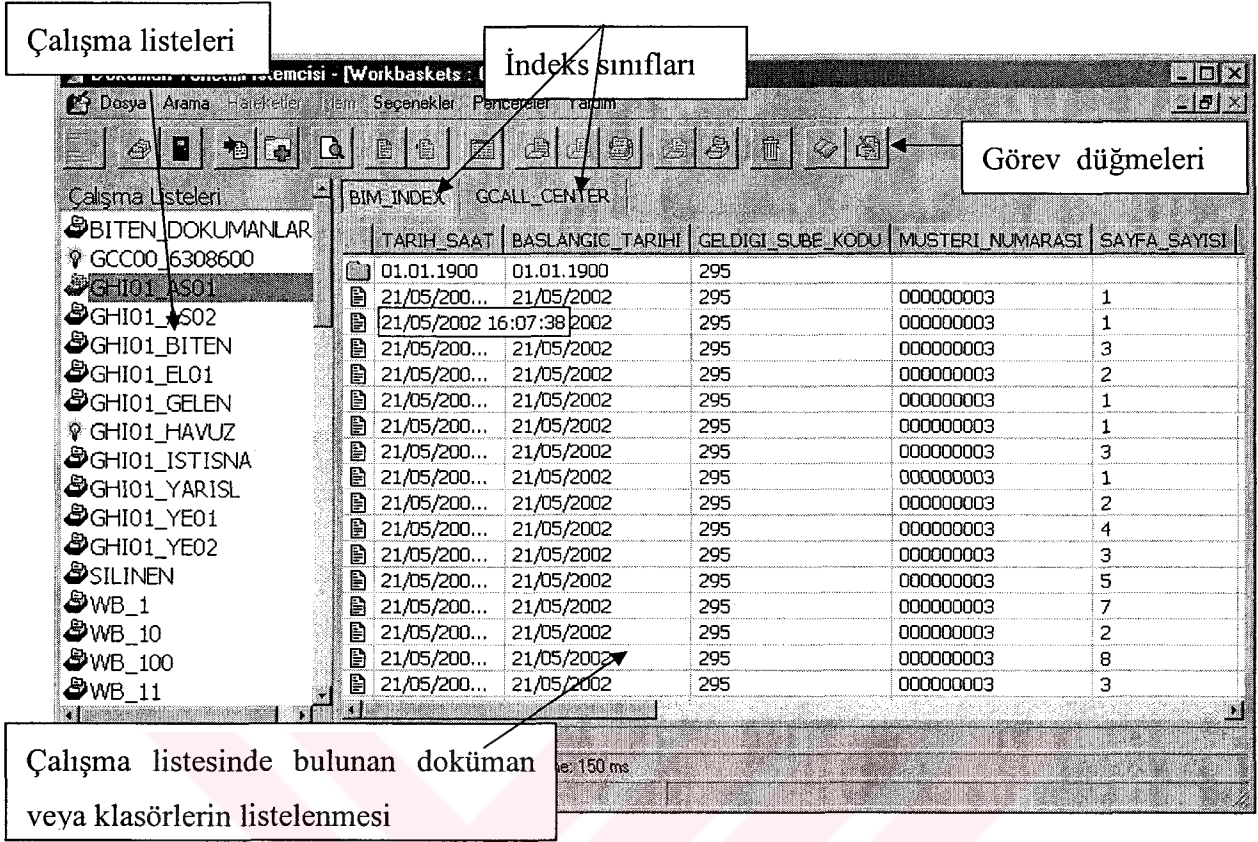
Son tuş ise düğme görüntüye not eklemek için kullanılacak düğme takımının açılabilmesini sağlayan düğmelerdir.

Bu düğme takımları standart istemci tarafından kullanılan düğme takımlarıdır. Buna ek olarak eğer standart istemci eğer bir excel dokümanını kendi içinde açarsa excelle ilgili olan tuş takımları da standart istemcinin tuş takımlarından biri haline gelir. Bu durumda düğme takımlarının ne işe yaradığını bulabilmek için excel uygulamasının yardımıyla yararlanılabilir.

6.2 Çalışma Listelerinin Listelenmesi

Çalışma listeleri dokümanları veya klasörleri bir iş akışına sokabilmek için kullanılan iş kuyruklarıdır.

Standart istemciye giriş yapıldığında Şekil 6.1’de anlatılan düğmelerden çalışma listelerini listeleyen düğmeye basıldığında kullanıcının görme yetkisine sahip olduğu kuyruklar çalışma listesi panelinde sıralanır. Eğer bunlardan biri çift tıklandığında aşağıdaki gibi bir çalışma listesini listelemiş olur.



Şekil 6.3 Çalışma listeleri ve detayları

Dikkat edilirse bir çalışma listesinin içerisinde birden fazla indekste indekslenmiş doküman mevcut olabilir. Yukarıdaki şekilde görüldüğü gibi hem BIM_INDEX sınıfında hem de GCALL_CENTER indek sınıfları içerisinde dokümanlar vardır.

Çalışma listeleri sadece dokümanları değil klasörleri de iş akışına sokabilirler. Şekil 6.3 de buna da bir örnek verilmiştir. Doküman veya klasörler çalışma listelerinde yer alabilirler.

6.3 Arama

Arama yapmak doküman yönetim sisteminde en geliştirilmiş fonksiyonlardan bir tanesidir. Doküman yönetim sistemlerinin en büyük avantajlarından bir tanesi aranan doküman veya klasöre anında ulaşabilmektir.

Arama sadece bir indeks sınıfında değil, tüm indeks sınıflarında yapılabilir. Bu durumda istemci uygulaması sistemde tanımlı olan ve kullanıcının arama yetkisi bulunan indeks sınıfları üzerinde verilen kriterlerle arama yapacak ve sonuçta bulduğu tüm nesneleri kullanıcıya getirecektir.

Şekil 6.4’de buna bir örnek verilmiştir. Arama tüm indeks sınıflarında başlangıç tarihi 19.05.2002 için yapılacaktır. Sistem içinde başlangıç tarihi kolonu bulunan tüm indeks sınıflarında eğer bu indeks sınıflarında kullanıcının yetkisi varsa arama yapacak, bulduğu sonuçları bir araya getirerek sonucu kullanıcıya gösterecektir.

Şekil 6.4 Tüm indekslerde arama yapma

Arama yapılmış ve ayrı indeks sınıflarda iki ayrı doküman bulunmuştur. Bu Şekil 6.5’de gösterilmiştir. Bu dokümanlardan birisi BIM_INDEX’te indekslenen diğeri ise GCALL_CENTER da indekslenen dokümanlardır. Bir doküman aynı anda sadece bir indekste olabilir. Bu durumda tüm sistemde başlangıç tarihi 19.05.2002 indeks bilgisiyle indekslenmiş sadece iki doküman vardır.

Arama sadece bir indeks sınıfında da yapılabilir. Şekil 6.6 de buna bir örnek verilmiştir. 295 kodlu şubeden gelen dokümanlar aranmaktadır. Ancak “Hepsi” seçeneği ile bu indeks sınıfı bilgilerine sahip olan klasörlerde işaretlenmiştir. ARA düğmesine basıldığında arama sonucunda bulunacak nesneler Şekil 6.7’de gösterilen biçimde kullanıcının önüne getirilecektir.

Şekil 6.6 Tek bir indeks sınıfında arama

TARIH_SAAT	BASLANGIC_TARIHI	GELDigi_SUBE_KODU	MUSTERİ_NUMARASI	SAYFA_SAYISI
27.05.200...	21.05.2002	295	31012312	
17.05.200...	17.05.2002	295	NULL	3
20.05.200...	15.05.2002	295	NULL	1
15.05.200...	15.05.2002	295	NULL	1
15.05.200...	15.05.2002	295	NULL	1
15.05.200...	15.05.2002	295	000000001	4
15.05.200...	15.05.2002	295	NULL	1
15.05.200...	15.05.2002	295	NULL	1
20.05.200...	15.05.2002	295	NULL	1
20.05.200...	15.05.2002	295	NULL	1
20.05.200...	15.05.2002	295	NULL	1
15.05.200...	15.05.2002	295	NULL	1
20.05.200...	15.05.2002	295	NULL	1
20.05.200...	15.05.2002	295	NULL	1
15.05.200...	15.05.2002	295	NULL	1
15.05.200...	15.05.2002	295	NULL	1
15.05.200...	15.05.2002	295	NULL	1
15.05.200...	15.05.2002	295	NULL	1

Şekil 6.7 Arama sonuçları

6.4 Klasör Yaratılması

Klasör yaratmak için indeks sınıfını seçmek ve indeks bilgilerini seçip YARAT demek yeterli olacaktır. Böylece sistemde boş bir klasör yaratılmış olacaktır. Tabi bunu yapacak kullanıcının o indeks sınıfı için klasör yaratma yetkisinin olması gerekmektedir.

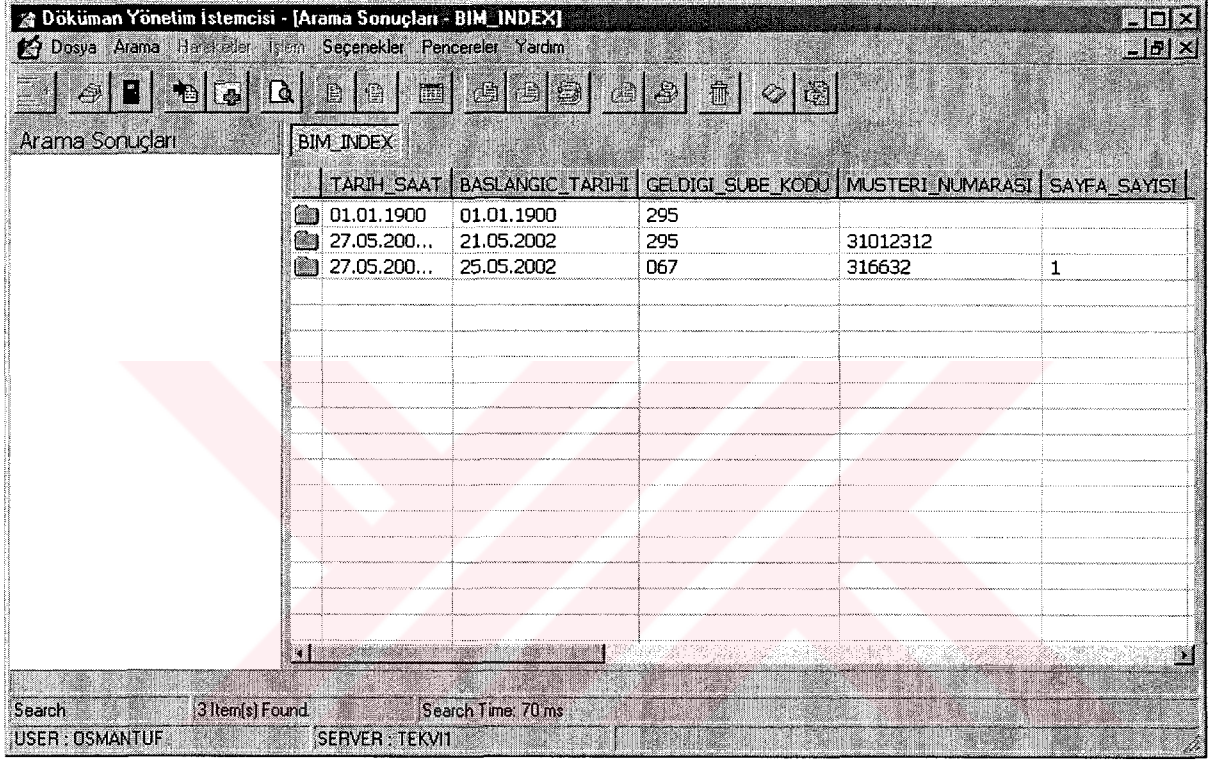
Şekil 6.8 Klasör yaratma

Yukarıdaki şekilde buna örnek verilmiştir. BIM_INDEX sınıfı içinde bir klasör yaratılmak için alanlar doldurulmuştur. Bu BIM_INDEX'te boş bir klasör yaratacaktır.

6.5 Klasörlerin Listelenmesi

Klasör yapısı, doküman yönetim sistemlerinin önemli avantajlarından bir tanesidir. Dokümanlar veya klasörler bir başka klasör içine yönlendirilebilirler. Ayrıca, bir doküman veya klasör farklı klasörlere de yerleştirilebilirler.

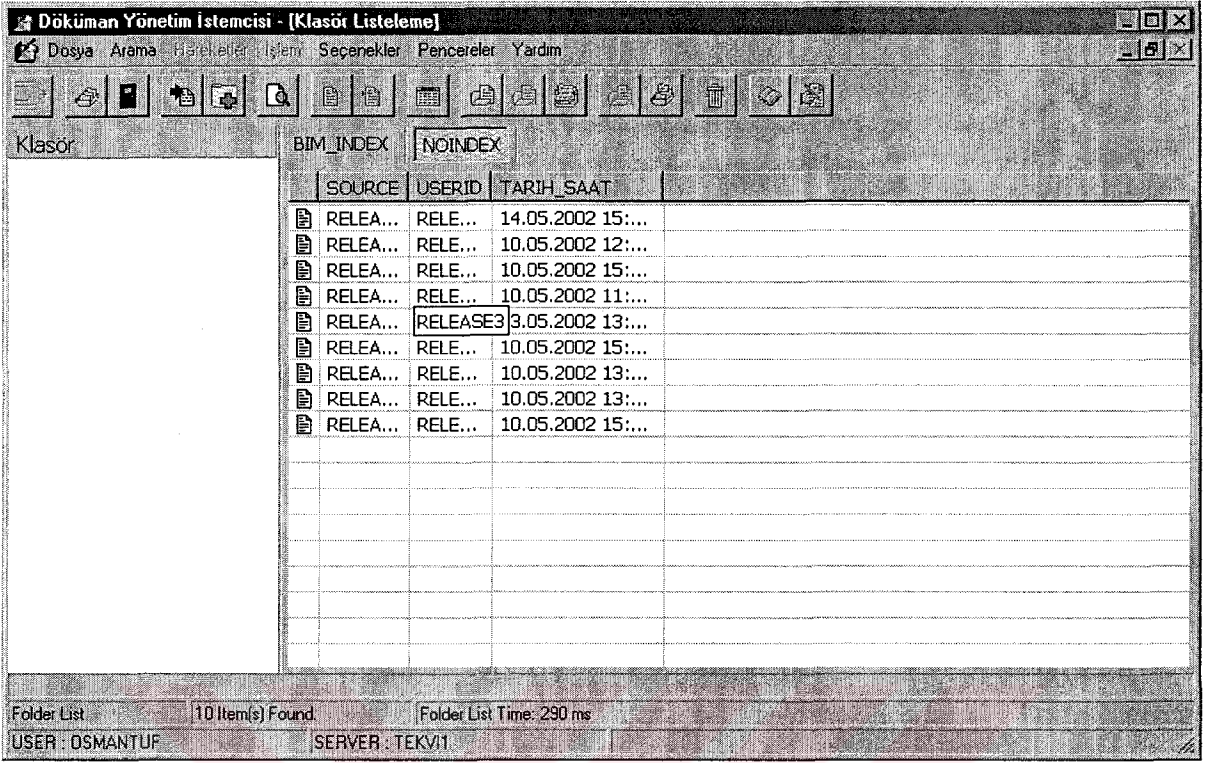
Klasörleri listelemek için klasörü bulmak ve çift tıklamak yeterli olacaktır. Klasörlerin listelenmesi çalışma listelerinin listelenmesi gibidir.



Şekil 6.9 Klasör bulma

Görüldüğü gibi klasörlerde indeks sınıflarında listelenebilmektedir. Bu özellikleriyle fiziksel doküman sistemdeki klasör mantığından hemen ayrılırlar.

Klasörleri listelemek için üzerinde çift tıklamak yeterli olacaktır. Şekilde görüldüğü gibi klasörlerin içindeki doküman veya diğer klasörler farklı indeks sınıflarında listelenebilirler. Zaten genellikle sistem tasarlanırken klasörleri listelemek için farklı bir indeks sınıfı, dokümanları listelemek için farklı bir indeks sınıfı yaratılır. Yani klasörlerin olduğu indeks sınıflarında dokümanları, dokümanların olduğu indeks sınıflarında dokümanları barındırmak içerik açısından doğru değildir. Ama istenirse sistem buna müsaade eder.



Şekil 6.10 Klasör listeleme

Burada da listelenen dokümanların üzerlerine çift tıklanarak dokümanlar açılabilir ve güncellenebilirler.

6.6 İmaj Dosyaların Gösterilmesi

Standart istemcinin gerçekleştirdiği en önemli görevlerden bir tanesi de Windows 2000 Kodak Imaging bileşenlerini kullanarak TIF,GIF,JPG gibi dosyaları kendi bünyesi içinde yönetebilmesidir. Özellikle TIF imajlarının gösterilmesi, değiştirilebilmesi ve güncellenebilmesi mümkündür.

Şekilde sisteme dışarıdan gelen bir faks gösterilmektedir. Kullanıcı bu faksı ya arama sonucu, ya bir çalışma listesinden yada klasörden bulmuş ve açmıştır. Kütüphane sunucuda TIF dosyasının genellikle standart istemci tarafından yönetilmesi konfigüre edilmiştir. Bu yüzden standart istemci yapısında bulundurduğu bileşenlerle bu imaj dosyasını açar ve kullanıcıya gösterir.

Doküman Yönetim İstemcisi - [Image Viewer]

Dosya Hareketler Arama Pencereleer Deęiřtme Görme Sayfa Yakınlař/Uzaklař Notlar Yardım

17-APR-2002 17:52 FROM OSMANTUF HARBIYE TO 120 P.81
CUMHURİYET İÇİřİ BAKANLIđI
KİMLİK BİLGİLERİ

ÜYE İřYERLERİ BAřVURU FORMU
(FORMDAKİ TÜM BİLGİLERİ LÜTFEN EKSKİZ VE BEYÖK HARFLERLE DOLDURUNUZ.)

İř YERİ TİCARİ ÜNVANI (Ticaret Sicil / (veya) oda kaydına göre)

İşyeri Adı: GEZGİN TURİZM İřL SAN VE TİC L TD řTİ
Sermaye: 5750 Şehir:
Ülke: Posta Kodu:
Tel.No1: Tel.No2:
Faks No: E-mail: (varsa yazılacaktır.)
Vergi Dairesi: Vergi No:

Tüzel Kişilik: ☐ Adi Şirket ☐ A.Ş. ☒ LTD ☐ Dernek/Vakıf
Oda Kayıt Bilgileri: (Ticaret sicil bilgileri varsa doldurulmayacaktır.)
Herhangi bir oda/derneęe kayıtlı iseniz adı:
Kayıt Tarihi: Kayıt No:
Ticaret Sicil Gazetesi Bilgileri: (Oda Kayıt bilgileri varsa doldurulmayacaktır.)
Ticaret Sicil Kayıt Tarihi: ____/____/____ Ticaret Sicil No:
Kuruluş Yılı: Şube/Bayii Sayısı: Biletin Serisi:

CONTENT TYPE: TIF PART NO: 0 VERSION NO: 0 (LAST)
USER: OSMANTUF SERVER: TEKVİİ

Şekil 6.11 Dokümanların gösterilmesi

Bu form sayesinde kullanıcı imaj üzerinde bir çok deęişiklikler yapılabilir ve dokümana kaydedilebilir. Artık doküman güncellenmiştir. Yine istenirse bu panel üzerinde bir çalışma listesine yönlendirme veya bir klasöre koyma gibi hareketlerde yapılabilir.

Doküman Yönetim İstemcisi - [Image Viewer]

Dosya Hareketler Arama Pencereleer Deęiřtme Görme Sayfa Yakınlař/Uzaklař Notlar Yardım

1

BİLGİ FORMU

İşlem Şube	Garanti Harbiye Şubesi
ANS NO	078.005790297.BD.0013273
No	5790297
Adı	GEZGİN TURİZM İřL SAN VE TİC L TD řTİ
Şubası	BD-Bilgi Giriř / Dięer
İmat Gereęi	Hayır
İşlem Sayısı	1
İliřkili İşlem	Hayır
Öncelik	Normal
Masraf/Komisyon	
Borçlu Hesap No	
Alacaklı Hesap No	
Tutar	NOTLAR
EFT Tutarı	
Kur Referansı	
Doęum Tarihi	
Baba Adı	

CONTENT TYPE: TIF PART NO: 0 VERSION NO: 0 (LAST)
USER: OSMANTUF SERVER: TEKVİİ

Şekil 6.12 Dokümanların güncellenmesi

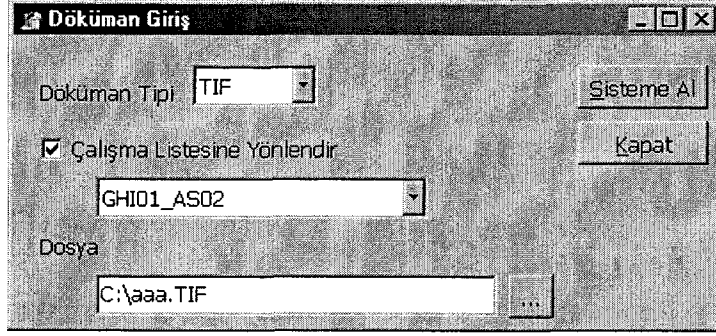
6.7 Sisteme Doküman Eklenmesi

Sisteme doküman eklemek için ařağıdaki şekildeki panel kullanılır. Doküman tipinde hangi

tipde bir doküman ekleyeceğimiz seçilir. Bu paneldeki doküman tipleri kütüphane sunucusunun DMSCONTENTCLASSES tablosunda yazılan doküman tipleridir.

Eğer içeri alınacak dokümanı bir çalışma listesine yönlendirmek istiyorsak bunu yönlendireceğimiz çalışma listesinin adını seçerek yapabiliriz. Bu çalışma listeleri sadece bizim görebildiğimiz çalışma listeleridir.

Dosya ise içeri alacağımız dosyanın fiziksel yeridir.

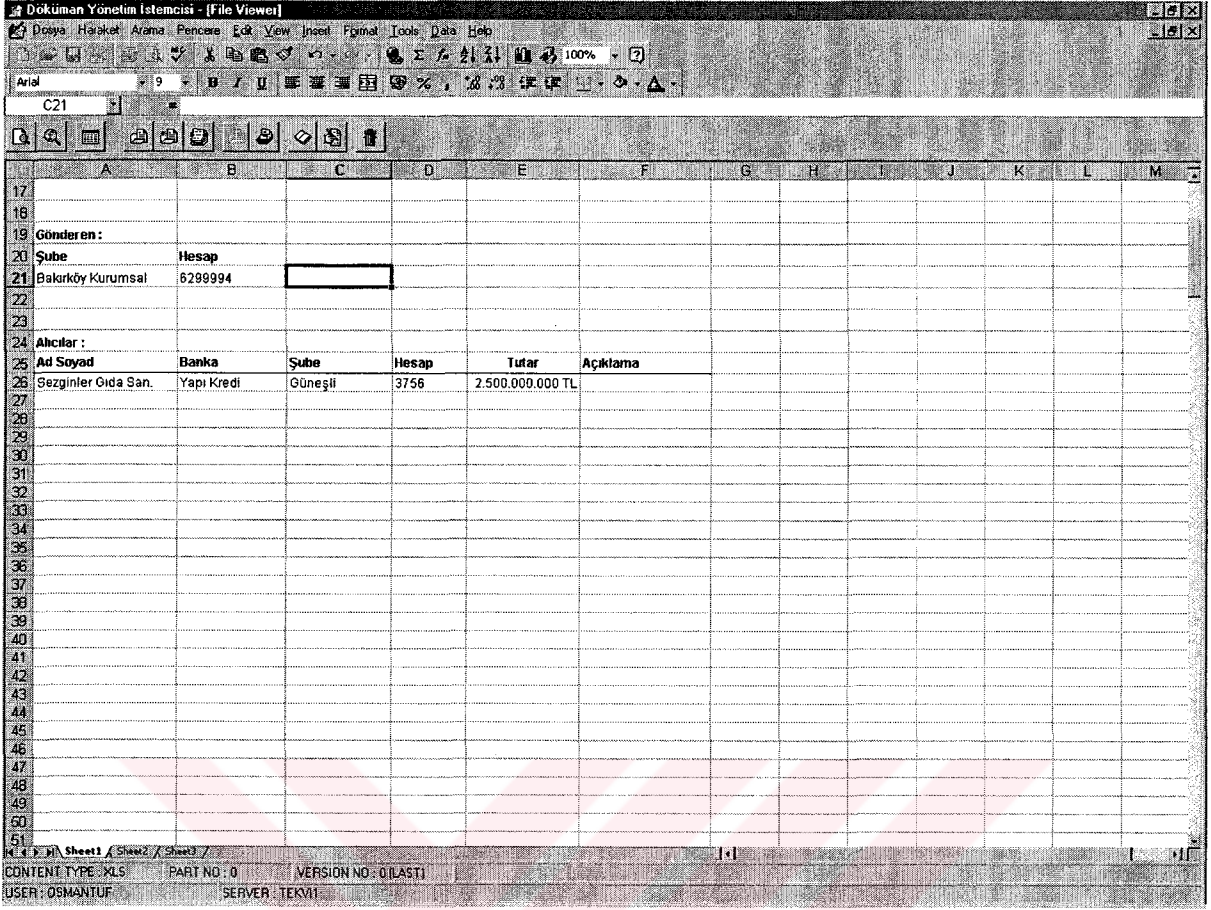


Şekil 6.13 Sisteme doküman ekleme

6.8 Farklı Tipte Dosyayı Uygulama İçinde Açma

İmaj dosyaları dışındaki doküman tipleri uygulama tarafından ya ayrı bir program çalıştırılarak yada kendi içinde başka bir programı çalıştırarak yönetilebilir.

İstemcinin bir doküman tipini nasıl yöneteceği kütüphane sunucuda konfigüre edilmiştir. Genellikle Word ve Excel dokümanları uygulamanın kendi içinde başka bir program vasıtasıyla açtığı (embedding) dokümanlardır. Aşağıdaki şekilde uygulama bir xls (Excel) dokümanı ile karşılaşmış ve kütüphane sunucusunda bu doküman tipini açmak için “uygulama içinde aç” parametresi konfigüre edilmiş olsun. Bu durumda aşağıdaki gibi excel programını uygulamanın kendi içinde açma imkanı vardır. Tek gerek ve şart uygulamayı açacak excel programının bilgisayarda yüklü olmasıdır.



Şekil 6.14 Bir excel dokümanın açılması

Yine bu paneldeyken araç çubuğunu kullanarak doküman yönetim sistemindeki işlemlerden herhangi birisi yapılabilir.

6.9 Farklı Tipte Dosyayı Uygulama Dışı Bir Programla Açma

İstemci “uygulama dışında aç” parametresi ile konfigüre edilmiş doküman tiplerini açabilmek için o doküman tipini açabilecek programı ayrı bir proses olarak çalıştırması gerekmektedir.

Aşağıdaki örnekte bir MP3 müzik dosyası görmekteyiz. Bir müzik dosyası sisteme daha önceden kaydedilmiş ve kullanıcı bu dokümanı bulup üstüne çift tıkladığında doküman yönetim sistemi işletim sisteminde yüklü olan MP3 oynatıcıyı bularak müzik dosyasının çalınmasını sağlar.



Şekil 6.15 Bir MP3 dosyasının açılması

Dikkat edilirse bu dokümanda araç çubuğu kullanılarak doküman yönetim sistemi içinde hareket ettirilebilir.

7. YÖNETİCİ PROGRAM

Yönetici program kütüphane sunucunun katalog bilgisinin hazırlanmasında ve güncellenmesinde kullanılan programdır.

7.1 Yönetici Program Nedir?

Doküman yönetim sisteminin kalbi olan kütüphane sunucu ve dokümanların saklanması için kullanılacak doküman sunucuların konfigürasyonu “Yönetici Program” adını verdiğimiz program tarafından yapılır.

Yönetici programın 10 paneli 10 ayrı iş yapar.

- 1) Anahtar alanların tanımlanması
- 2) Ayrıcalık setlerinin oluşturulması
- 3) Giriş kontrol listelerinin tanımlanması
- 4) Kullanıcıların tanımlanması
- 5) Grupların yönetilmesi
- 6) İndeks sınıflarının oluşturulması
- 7) Çalışma litelerinin tanımlanması
- 8) Sistem konfigürasyonun yapılması
- 9) Doküman sunucuların belirlenmesi ve disk parçalarının oluşturulması
- 10) SMS Seçeneklerinin tanımlanması

Bunlara geçmeden önce bu kavramları bir kez defa detaylandırmakta fayda var.

7.2 Anahtar Alanlarının Tanımlanması

Anahtar alanlar, veritabanında indeks sınıfları oluşturacak belli bir veri tipi olan kolon adlarından başka bir şey değildirler. Mesela bir Musteri_Ad (varchar(20)) tipik bir anahtar alan ismidir.

Anahtar alanlar topluluğu indeks sınıfları tablolarını oluşturmak için kullanılırlar. İndeks sınıfları tanımlanırken indeks sınıflarının içinde yer alacak nesneler için veritabanı tablosunu oluşturacak bazı alanlara ihtiyaç vardır. O alanlar bu anahtar alanlar içinden seçilir.

Herhangi bir anda farklı bir anahtar alan sisteme eklenebilir. Ancak bir anahtar alanın

minimum ve maksimum uzunluklarıyla veya veri tipiyle oynamak tanımlandıktan sonra mümkün değildir.

Eğer bu anahtar alan bir indeks sınıfında bir kolon olarak kullanılmaya başlanmışsa bu alanı silmek mümkün değildir.

Bunun için anahtar alanların tanımlanmasında iyi ve bir kere düşünmekte fayda vardır.

7.3 Ayrıcalık Setleri

Ayrıcalık setleri bir tür yetki topluluğudur. Bir indeks sınıfından doküman veya klasör okuma, doküman veya klasör silme, bir indeks sınıfına doküman veya klasör ekleme, bir çalışma listesindeki bir dokümanın öncelik bilgisini değiştirme gibi güvenliği ilgilendiren tüm kavramlar ayrıcalık setlerinin içinde gizlidirler.

Ayrıcalık setleri Giriş Kontrol Listelerinin oluşturulmasında kullanılırlar. Giriş kontrol listeleri oluşturulurken iki şeye ihtiyaç vardır. Bunlardan birincisi kullanıcı veya grup, diğeri ise ayrıcalık setidir. Bu iki kavramı bir araya getirirsek, Giriş Kontrol Listelerinin manasınıda bulmuş oluruz.

Ayrıcalık setlerinin içinde geçen tüm ayrıcalık bilgileri detaylı bir şekilde anlatılacaktır.

7.4 Giriş Kontrol Listeleri

Giriş Kontrol Listeleri en doküman yönetim sisteminin en önemli parçasıdır. Çünkü birebir sistemin güvenlikle ilgilidirler.

Giriş Kontrol Listelerinin içerisinde 3 çeşit kayda rastlamak mümkündür.

1. Kullanıcı-Ayrıcalık Seti
2. Grup-Ayrıcalık Seti
3. PUBLIC-Ayrıcalık Seti

Bunlardan birincisi, bir kullanıcının o giriş kontrol listesi içinde hangi ayrıcalık seti ile tanımlandığını gösterir. İkincisi ise bir grubun o giriş kontrol listesi içinde hangi ayrıcalık seti ile tanımlandığını belirtir. Üçüncüsü o giriş kontrol listesi için genel bir ayrıcalık seti tanımlar. Eğer bir giriş kontrol listesinin içinde PUBLIC grubuyla bir ayrıcalık seti tanımlanmışsa başka kayıtlara giriş izni verilmez. Yani tüm kullanıcılar o giriş listesine bağlı çalışma listeleri veya indeks sınıflarına tanımlı olan ayrıcalık seti ile girerler.

Giriş kontrol listeleri sistemde iki nesneye bağlanır.

- a. İndeks Sınıfları
- b. Çalışma Listeleri

Görüldüğü gibi bir indeks sınıfı veya çalışma listesi tanımlandığında bununla beraber onlara bağlı bir giriş kontrol listesi tanımlanmak zorundadır. Bu da sistemdeki kullanıcıların bu indeks sınıfları veya çalışma listeleri üzerindeki haklarını otomatik olarak tanımlama anında belirler.

Burada dikkat edilmesi gereken bir giriş kontrol listesi içinde hem kullanıcıların hemde grupların ayrıcalık setleriyle tanımlanacak olmalarıdır. Çünkü bir kullanıcı hem kendi hemde tanımlı olduğu gruplardan dolayı iki veya daha fazla ayrıcalık setine sahip olabilir. Bu durumda sistem o kullanıcıyı tanımlandığı giriş kontrol listesi içinde bu ayrıcalık setlerini “OR”layarak oluşan maksimum ayrıcalıkla değerlendirecektir.

Mesela;

ACL_1 adlı bir giriş kontrol listesi

PUBLIC MAXPRIV

kaydından oluşsun. Bunun anlamı tüm kullanıcıların, bu giriş kontrol listesine bağlanarak tanımlanan indeks sınıflarında veya çalışma listelerinde MAXPRIV ayrıcalık setindeki ayrıcalıklarla çalışacaklarıdır.

Mesela;

ACL_1 isimli giriş kontrol listesi

OSMANTUF USERPRIV

UFUKY MAXPRIV

gibi iki kayıttan oluşsun. Buna göre ACL_1 isimli giriş kontrol listesine bağlanarak tanımlanan indeks sınıflarında veya çalışma listelerinde OSMANTUF USERPRIV ayrıcalık setindeki ayrıcalıklarla, UFUKY ise MAXPRIV ayrıcalık setindeki ayrıcalıklarla çalışacaktır.

Şimdi biraz daha karmaşık bir durumu örnek verelim;

OSMANTUF GRP_1 ve GRP_2 gruplarının içinde olsun;

MAXPRIV 00110100101000010100101 gibi bir ayrıcalık koduna sahip olsun.

USERPRIV 01100010100001000100100 gibi bir ayrıcalık koduna sahip olsun.

Ve;

ACL_1 isimli giriş kontrol listesi

GRP_1 USERPRIV

GRP_2 MAXPRIV

Bu durumda OSMANTUF GRP_1 grubu içinde olduğundan USERPRIV haklarına sahip olacaktır. GRP_2 grubundanda MAXPRIV hakları gelir. Böylece kullanıcı bir giriş kontrol listesinde iki ayrıcalık setine sahiptir.

Bu durumda;

MAXPRIV 00110100101000010100101

USERPRIV 01100010100001000100100

OR

01110110101001010100101

ayrıcalık seti ile çalışmaya başlar. Buna maksimum hak verme diyoruz.

Ancak kullanıcının tek başına tanımlı olduğu bir kayıt olsaydı

GRP_1 USERPRIV

GRP_2 MAXPRIV

OSMANTUF OSMANPRIV

OSMANTUF gruplarından gelen ayrıcalık setlerine bakılmaksızın OSMANPRIV yetkisi ile çalışmak zorunda kalacaktır.

Kontrol Listelerinin oluşturulmasını matematiksel olarak formülize edersek;

Ayrıcalıklardan her birini a ile gösterirsek ve A 'ya ayrıcalık seti dersek;

$$A = a_1, a_2, a_3, \dots, a_n$$

ayrıcalık kümesini belirtsin. a_1, a_2, a_3 ayrıcalıklarının her biri ayrı hakları belirtsin.

Sistemdeki kullanıcılardan herbirini k ile gösterir ve K ya kullanıcı seti dersek;

$$K = k_1, k_2, k_3, \dots, k_n$$

Sistemdeki kullanıcı gruplarından herbirini r ile gösterir ve R ye grup seti dersek;

$$R = r_1, r_2, r_3, \dots, r_n$$

Giriş listesinin her bir elemanına g ile gösterir ve G 'ye Giriş Kontrol Listeleri dersek;

$$G = g_1, g_2, g_3, \dots, g_n$$

Ve;

$$g = \psi(A\chi + (K\chi \vee R\chi))$$

ifade edilebilir. Yani bir giriş kontrol listesi kayıdı bir Ayrıcalık Seti ve bir kullanıcı veya kullanıcı grubu ile ifade edilir.

Bu durumda;

$$G = \psi(\psi(A\chi + (K\chi \vee R\chi)))$$

Yani bir Giriş Kontrol Listesi Ayrıcalık ve Kullanıcı kayıtlarının setinden oluşmaktadır.

7.5 Kullanıcıların Tanımlanması

Kullanıcı tanımlarken tek bir değere dikkat etmek gerekir. Bu değer kullanıcı seviyesidir. Bu değer doküman yönetim sisteminin standart istemcisinde kullanılmamasına karşın, bu parametreyi kullanıp program geliştirecek kişiler için bir güvenlik parametresi olarak kullanılabilir.

Bir doküman bir indeks sınıfında listelenirken bir seviye kolonu ile beraber indekslenir. Kullanıcı seviyesi bu doküman seviyesinin altında kalanlar bu dokümanları göremezler. Bu bir tür programlama ve aynı indeks sınıfı içinde güvenlik getirme tekniğidir.

7.6 Grupların Yönetilmesi

Gruplar kullancılardan bir araya gelmesinden oluşan topluluklardır. Gruplar özellikle giriş kontrol listelerinin tanımlanmasında getirdiği kolaylıklardan dolayı yararlıdır.

Bir grup içinde başka bir grup olamaz.

7.7 İndeks Sınıfları

İndeks sınıfları dokümanların veya klasörlerin indekslendiği veritabanı tablolarıdır.

Bir indeks sınıfını tanımlamak istediğimizde indeks sınıfının adını ve hangi giriş listesine bağlamak istediğimizi belirtmemiz gerekir.

İndeks sınıfı sisteme tanımlandığında sistem bu indeks sınıfı için oluşturacağı tablo ismini kendi oluşturur. Bu isim STR prefiksine beş karakterlik bir sayının yazılmasıyla oluşturulur. Daha sonra sistem yöneticisi anahtar alanlar içinden bazılarını seçerek bu tabloya ekler ve indeks sınıfının tanımını yapmış olur.

7.8 Çalışma Listeleri

Doküman yönetim sisteminin en güçlü yönlerinden bir tanesi doküman veya klasörleri iş akışına sokabildiği çalışma listeleridir. Çalışma listeleri farklı indeks sınıflarındaki klasör veya dokümanları içinde listeleyebilecek şekilde tasarlanmıştır.

Çalışma listesi tanımlarken iki tipte tanımlama yapılabilir. Bunlardan bir tanesi liste alınabilecek çalışma listeleri, diğeri ise aynı anda sadece bir tane dokümanın gösterilebileceği hızlı çalışma listeleridir.

Hızlı çalışma listeleri yeni bir dokümanın sistem tarafından atanması demektir. Yani sistem bu tür çalışma listelerinde özellikle öncelik parametresinden yola çıkarak en öncelikli doküman veya klasörü isteyen istemciye döndürür.

Liste tipli çalışma listelerinde böyle bir özellik yoktur. Bu çalışma sepetleri sürekli listelenir ve içlerindeki tüm klasör ve doküman yapılarını gösterirler.

7.9 Sistem Konfigürasyonu

Sistem konfigürasyonu panelinde bazı performans ve kontrol parametreleri bulunur.

Doküman sunucuların üzerinde çalışan FTP sunuculara bağlanılabilmesi için FTP kullanıcı, şifre ve port değişkenleri buradan değiştirilebilir.

Ayrıca maksimum şifre deneme sayısı, bir nesnenin maksimum check-out zamanı, istemciye döndürülecek maksimum kayıt sayısı, eğer varsa logon mesajı burada verilir.

Sistemi veya sisteme girişı kilitleyecek parametrelerde burada deęiştirilebilir.

7.10 Doküman Sunucuları ve Disk Parçacıkları

Doküman sunucular bilindięi gibi dokümanlar saklanması ve istemcilere dağıtılması ile işleri üstlenen dosya sunuculardır.

Sistemin çalışabilmesi için en az bir doküman sunucu ve buna baęlı bir disk bir ve bu disk üzerinde bir veya birden fazla disk parçasının sisteme tanımlanması gerekir.

7.11 SMS Seçenekleri

SMS Sunucu tüm doküman yönetim sistemin disk yönetimini yapan merkezi bir programdır. Bu program bazı seçeneklerle çalıştırılabilir.

Mesela SMS Sunucusu aktif doküman sunucu üzerindeki aktif disk parçasını periodik olarak belli saatlerde kontrol edebileceęi gibi bunu her gün bir kere belli bir saatte gerçekleştirebilir. Bu durumu mailleyebilir veya maillemeyebilir.

Ayrıca doküman sunucular üzerinde çalışan FTP sunucularının ulaşabilirlik ulaşılamazlıklarını kontrol edebilir. Ulaşılamazlarsa bu durumu mailleyebilir.

Tabi bu maillerin kime gidecekleri hangi mail sunucu üzerinden gidecekleride detaylarda verilmelidir.

7.12 Sisteme Giriş

Sisteme giriş için kullanıcı kodunun doldurulup şifrenin verilmesi ve doęru kütüphane sunucusunun seçilmesi gerekir.

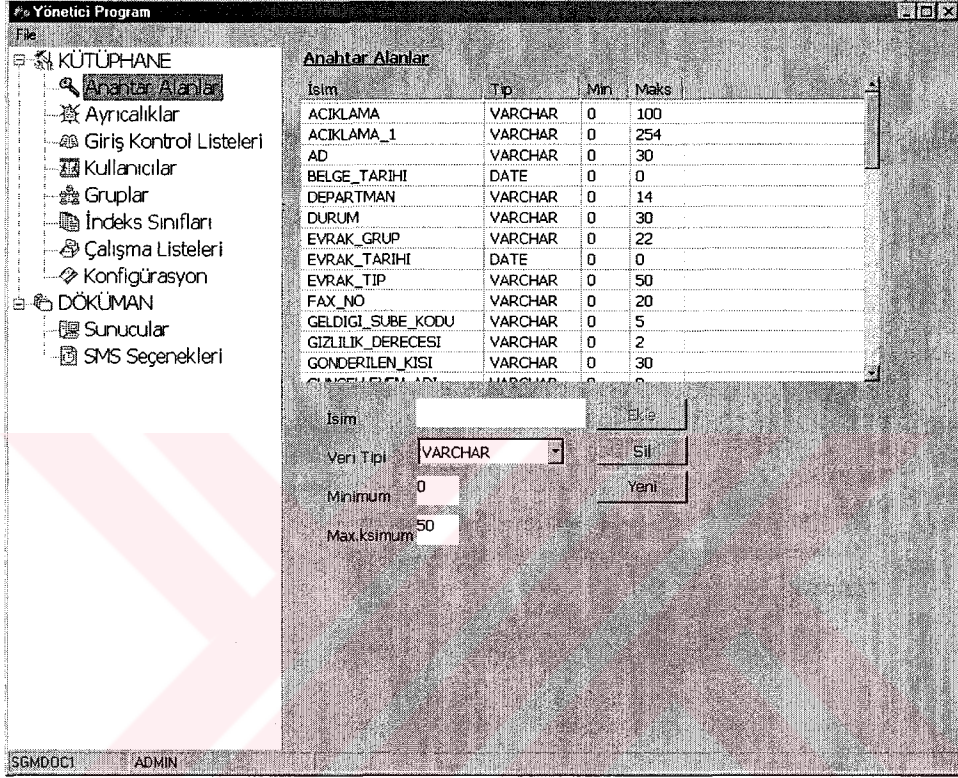
Şekil 7.1 Sisteme giriş ekranı

Kullanıcılara tanımları sırasında bu yönetici programları ile ilgili bazı yetkiler verilebilir.

Yönetici programa hangi kullanıcıyla giriliyorsa o kullanıcının yetkili olduğu paneller açılacaktır. ADMIN tüm panellere ulaşmaya yetkilidir.

7.13 Anahtar Alanlar

Anahtar alanlarının tanımlandığı paneldir.

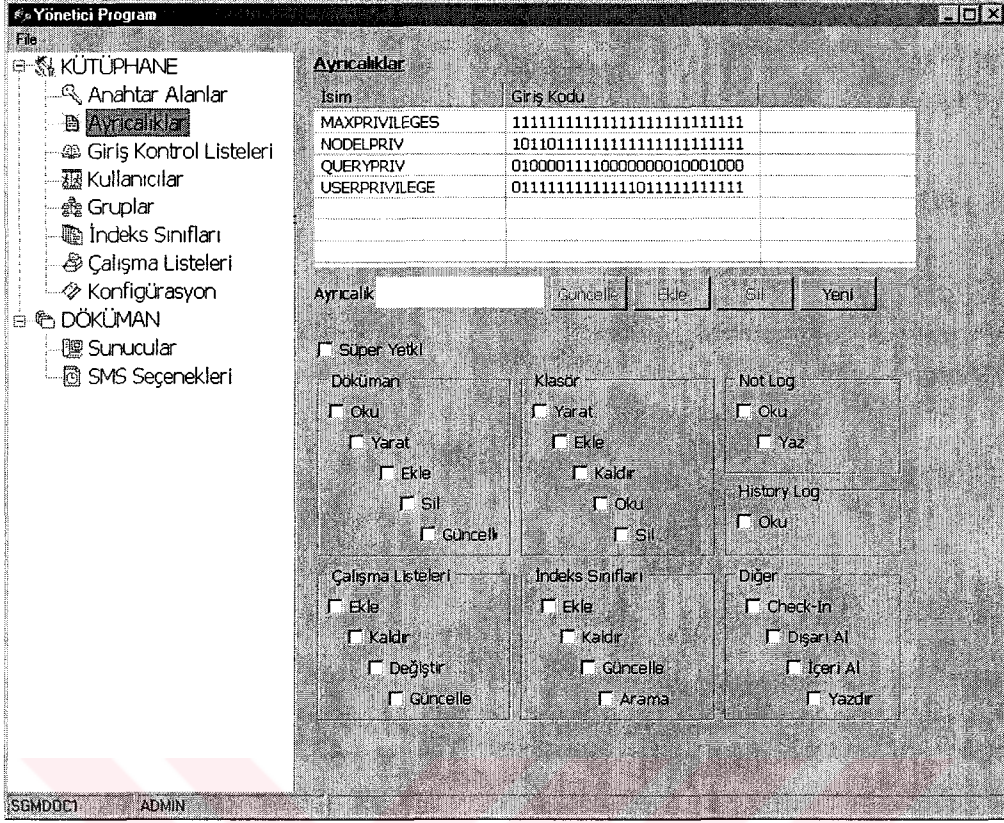


Şekil 7.2 Anahtar alanların tanımlanması

Anahtar alanlar tanımlanırken isminin, veri tipinin ve eğer VARCHAR ise minimum ve maksimum değerlerinin verilmesi gerekir.

7.14 Ayrıcalıklar

Ayrıcalık setlerinin tanımlandığı paneldir.

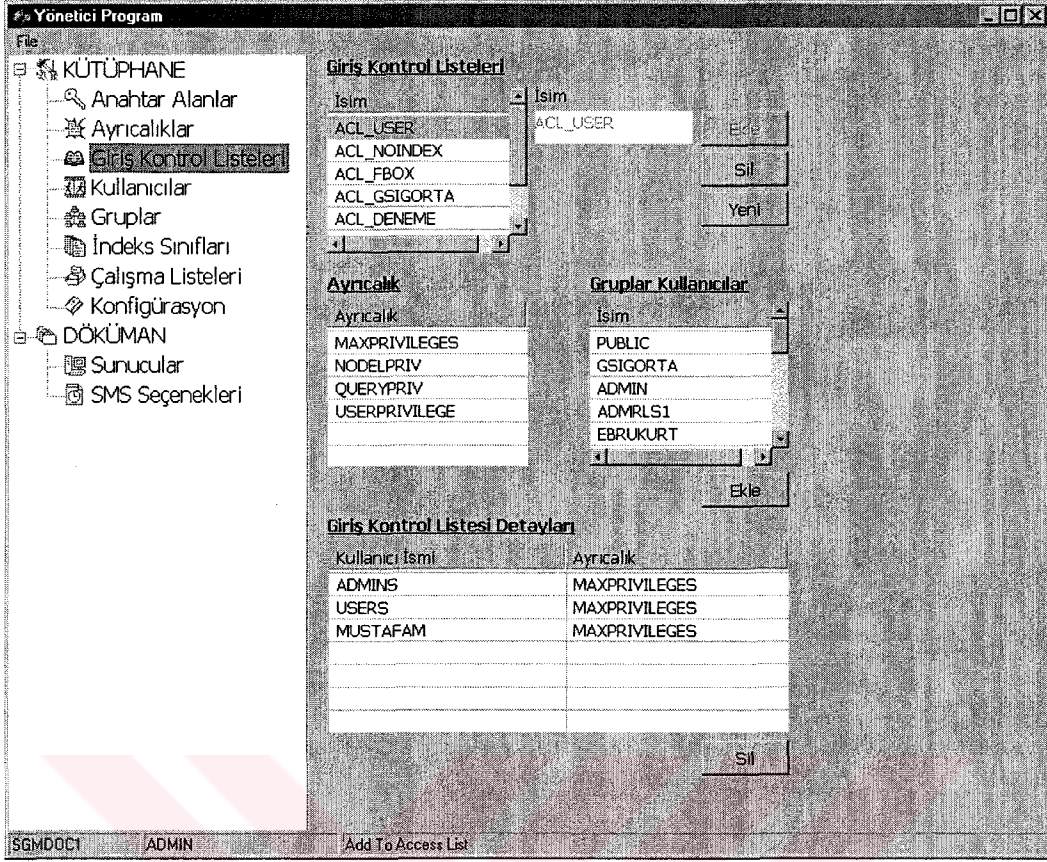


Şekil 7.3 Ayrıcalık setlerinin tanımlanması

Süper Yetki tüm yetkileri içerir. Bu yetkiye sahip olan tüm yetkilere sahip demektir. doküman okuma, yaratma, ekleme, silme, güncelleme, klasör yaratma, ekleme, kaldırma, okuma, silme tamamiyle indeks sınıfları bazındaki yetkililerdir. Aynı zamanda bir indeks sınıfına ekleme, bir indeks sınıfından kaldırma, bir indeks sınıfının güncellenmesi ve üzerinde arama yapılması yetkileride indeks sınıfı bazında yetkililerdir. Dışarı al, içeri al ve yazdır yetkileride indeks sınıfındaki yetkililerdir. Çalışma listesi altındaki ekle, kaldır, değiştir ve güncelle yetkileri çalışma listesi bazında yetkililerdir. NotLog yaz, oku ve history oku yetkisi kullanıcının indeks sınıfı bazındaki diğer yetkileridir.

7.15 Giriş Kontrol Listeleri

Giriş kontrol listelerinin tanımlandığı paneldir.



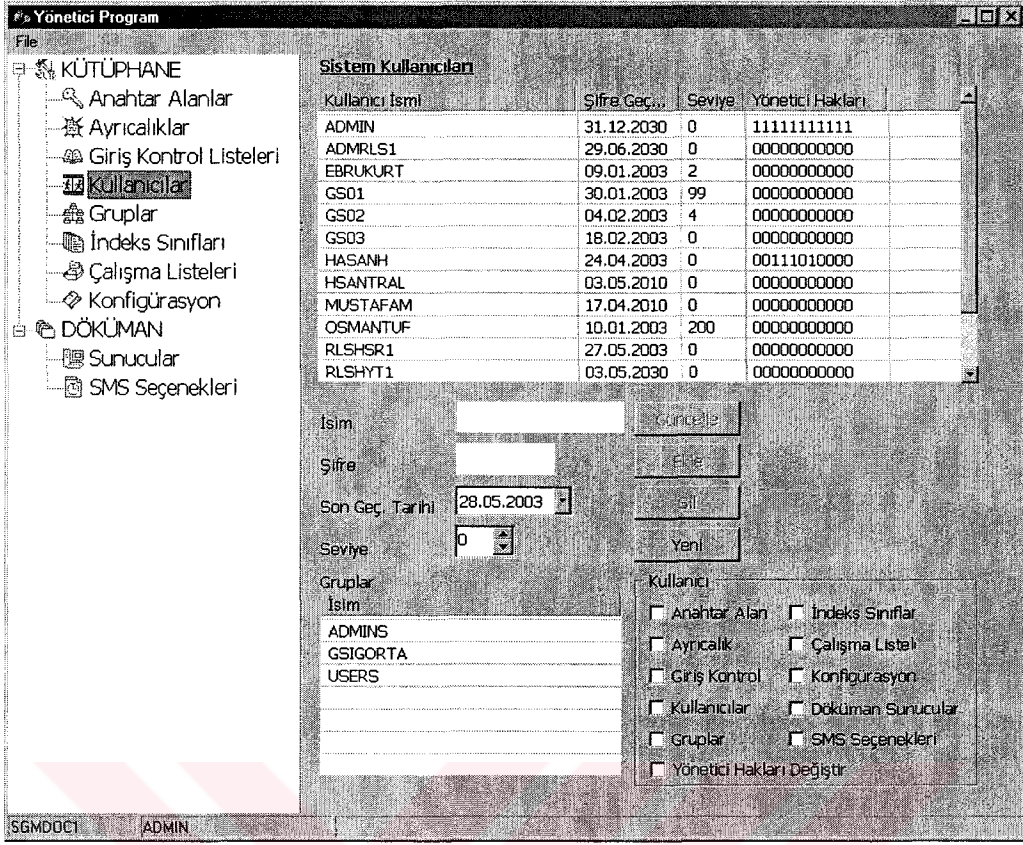
Şekil 7.4 Giriş kontrol listelerinin tanımlanması

Giriş kontrol listeleri ve bunların detayları buradan tanımlanırlar. Bir giriş kontrol listesi detayının oluşturulabilmesi için ilk önce o giriş listesi kaydı üzerinde çift tıklama yapıldıktan sonra, ayrıcalık bölümünden bir ayrıcalık ve gruplar, kullanıcılar bölümünden bir kayıt seçilip Ekle tuşuna basılmalıdır.

Eğer giriş kontrol listesinden bir kayıt çıkartılacaksa, giriş kontrol listesi detaylarından kayıtlar seçilerek Sil tuşuna basılır.

7.16 Sistem Kullanıcıları

Sistem kullanıcılarının tanımlandığı paneldir. Kullanıcı ismi, şifresi, şifrenin kullanılabileceği son zaman ve kullanıcı seviyesi işaretlenip kullanıcı bölümünden kullanıcının Yönetici Programında hangi yetkilerle çalışacağı belirlendikten sonra Ekle tuşuna basılarak kullanıcı eklenebilir.



Şekil 7.5 Sistem kullanıcılarının tanımlanması

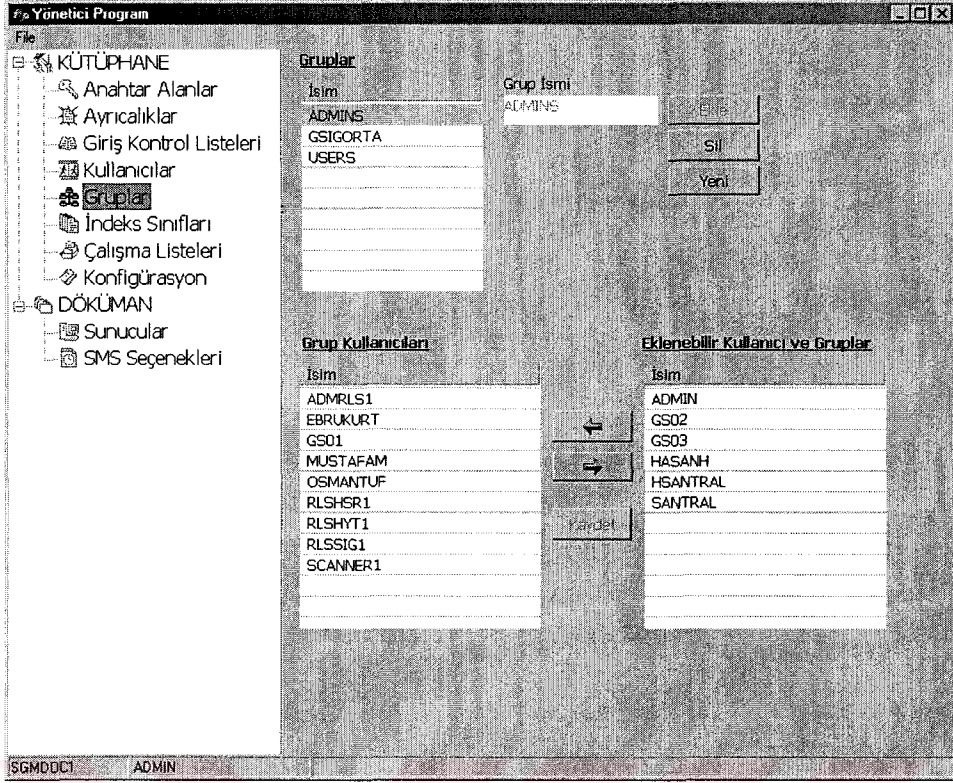
Görüldüğü gibi Kullanıcının yönetici programı için haklarının tanımlandığı bölümde panel isimleri verilmiştir.

Eğer bir kullanıcı kendi koduyla yönetici programa girerse hangi panellere izin verilmişse kullanıcıya o panel gözükür. Ayrıca ADMIN kullanıcısından başka hiçbir kullanıcı hiçbir durumda ADMIN kullanıcısının haklarını değiştiremez.

Eğer bir kullanıcıya Yönetici Hakları Değiştir seçeneğinde işaretlenmişse bu kullanıcı ADMIN den başka tüm kullanıcıların kullanıcısının yönetici programındaki haklarını değiştirebilir.

7.17 Gruplar

Grupların tanımlandığı paneldir. Sistem bir grup içine başka bir grubun girmesine veya bir kullanıcının 2 defa aynı grup içinde tanımlanmasına izin vermez. Zaten sağ taraftaki Eklenebilir Kullanıcı ve Gruplarda soldaki kullanıcılar gözükmez.



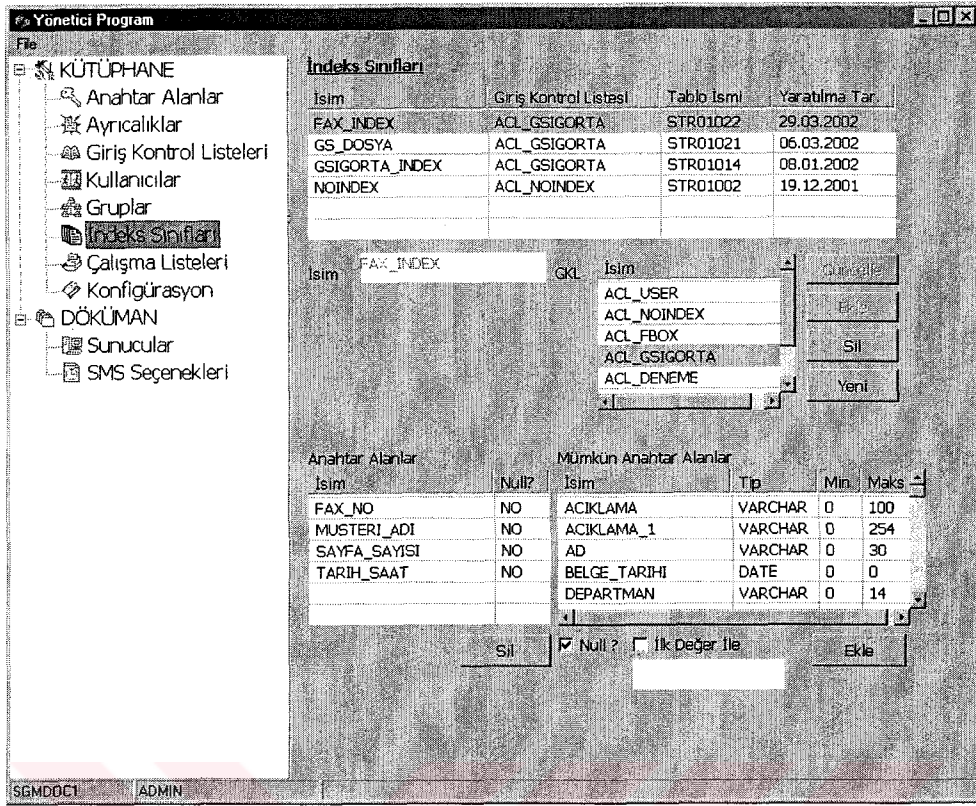
Şekil 7.6 Grupların tanımlanması

Yapılması gereken tek iş oklarla kullanıcıları ekleyip çıkartmak ve Kaydet tuşuna basmaktır.

7.18 İndeks Sınıfları

İndeks sınıflarının tanımlanması veya tanımlanan indeks sınıflarının güncellenmesini sağlayan paneldir.

İndeks sınıfının tanımlanması için GKL bölümünden bir giriş kontrol listesinin seçilmesi gerekmektedir. Daha sonra ekleye basılıp indeks sınıfı seçilerek Mümkün Anahtar Alanlar bölümünden anahtar alanlar seçilerek tabloya kolon olarak eklenebilir.



Şekil 7.7 İndeks sınıflarının tanımlanması

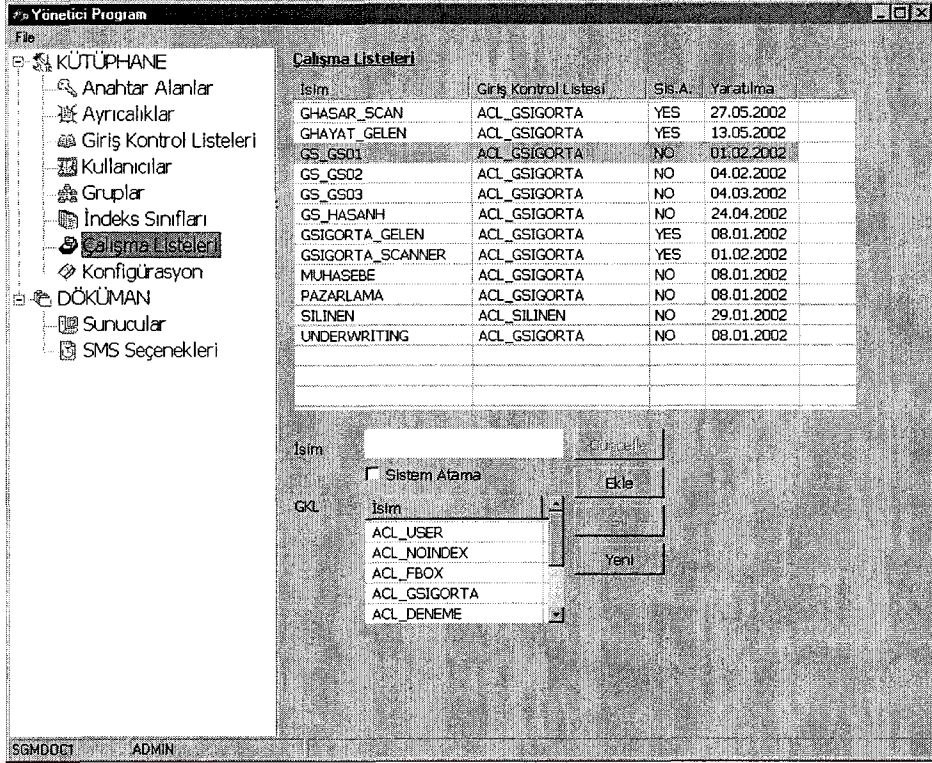
Bu kolonlar eklendikçe mümkün anahtar alanlardan çıkartılacaklardır. Bir kolon silmek için Anahtar Alanlar bölümünün altındaki Sil tuşu kullanılır.

Kolon eklerken NULL değere izin verilmesi veya ilk değerle atanması gibi seçenekleri işaretlemek kolonun tanımlanma parametrelerini değiştirecektir.

Bir indeks sınıfının tamamen silinebilmesi için o tablonun içinin boş olması gerekir. Veri tabanı tablosunun ismi Tablo İsmi kolonunda yazılıdır. Ancak silme işlemi asla veritabanının tablosundan yapılmamalıdır. Bu taktirde kütüphane sunucusunun bazı tablolarında gereksiz kayıtlar kalacaktır.

7.19 Çalışma Listeleri

Çalışma listelerinin tanımlandığı paneldir. Yapılacak iş bir isim vermek ve Hızlı liste olup olmayacağını seçmektir.



Şekil 7.8 Çalışma listelerinin tanımlanması

Tabi mutlaka bir Giriş Kontrol Listesi seçilmesi de gereklidir.

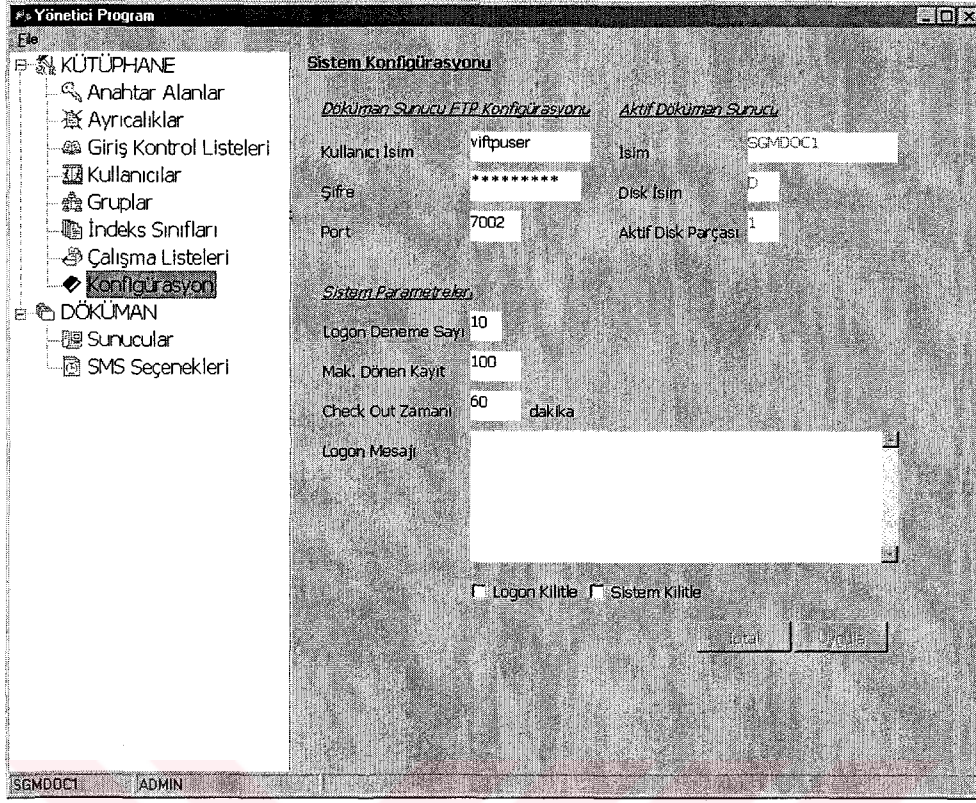
Bir Çalışma Listesini silebilmek için içindeki nesnelerin hepsinin silinmesi gerekir. Bu kesinlikle istemci nesnesinin objeleriyle yapılmalıdır. Direk veritabanından yapılacak silinmeler bazı tablolarda gereksiz nesneler bırakacaklardır.

7.20 Sistem Konfigürasyonu

Sistem parametrelerinin girildiği paneldir.

Bu panelde doküman sunucular üzerindeki FTP sunucularına giriş için gereken parametreler ve sistem performansını etkileyebilen bazı parametreler vardır.

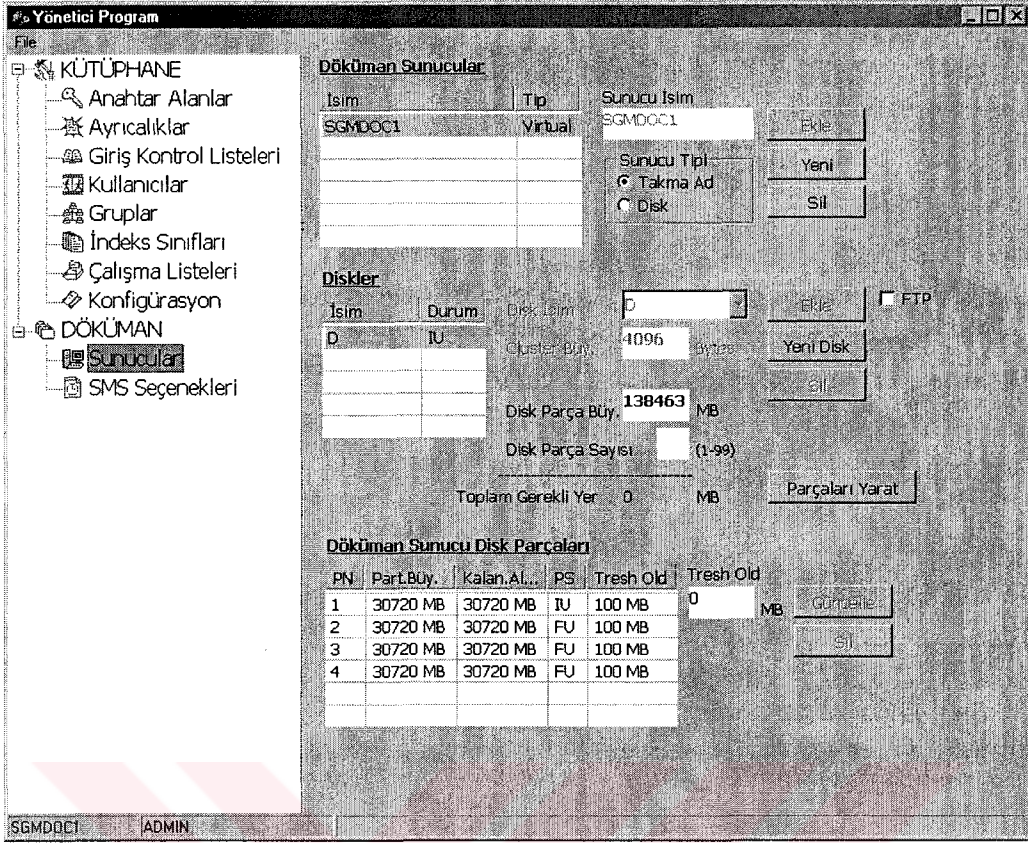
Aktif Doküman Sunucu bölümündede sadece okunabilecek değerler göze çarpar. Bunlar o anki aktif doküman sunucu, disk ve disk parçaçıdır.



Şekil 7.9 Sistem konfigürasyon paneli

7.21 Doküman Sunucuların Belirlenmesi ve Disk Parçalarının Oluşturulması

Doküman Sunucularının, disklerinin ve disk parçalarının tanımlandığı, silindiği, değiştirildiği paneldir. Tüm sistem disk konfigürasyonu buradan yapılabilir.



Şekil 7.10 Doküman sunucuların tanımlanması

Doküman sunucular bölümünden yeni bir doküman sunucu tanımlanabilir. Bir doküman sunucunun silinebilmesi için tüm disklerinin FU ('Gelecekte Kullanılacak') durumunda olması gerekmektedir.

Sunucu tanımlandıktan sonra üzerine çift tıklanarak disk konfigürasyonu yapılır. Diskler eklenir veya çıkartılır. Bir diskin silinebilmesi için kendi ve disk parçacıklarının durumu 'FU' olmak zorundadır. Aksi halde sistem bir diskin silinmesine izin vermez.

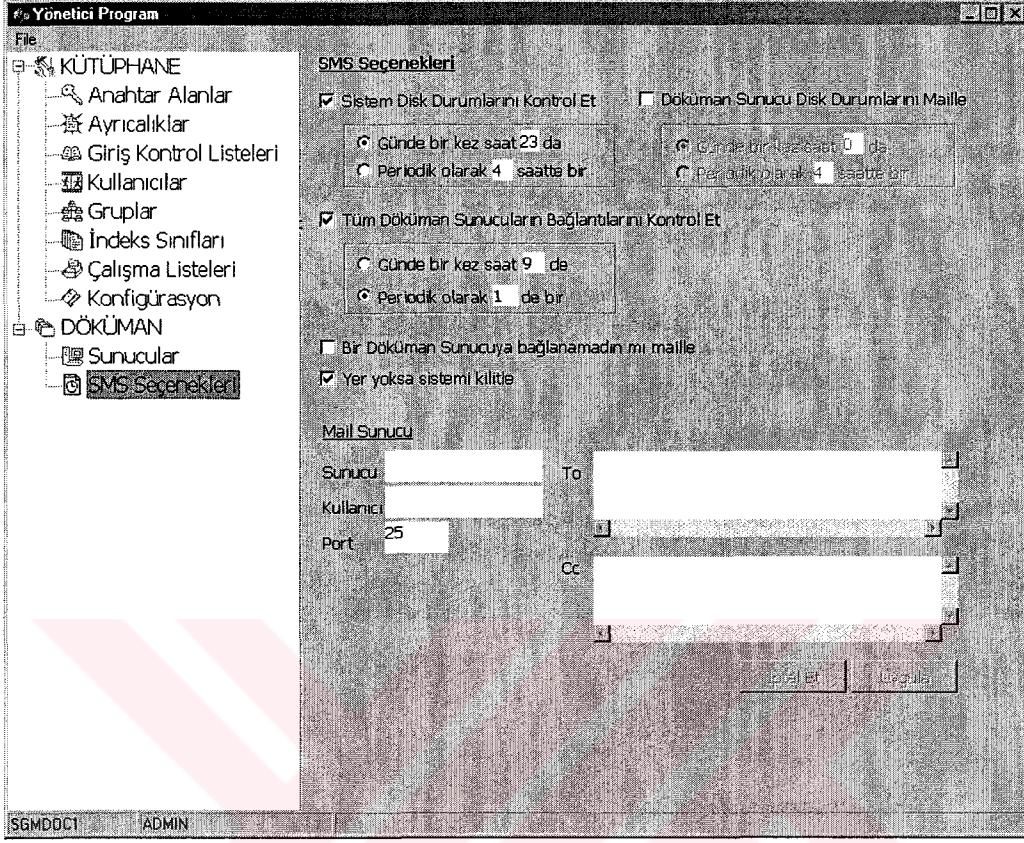
Son olarak diskin üzerine çift tıklanır ve diskin kaç alt parçaya bölüneceği ve bu bölünmeden hangi büyüklükte disk parçacıkları oluşturulmak isteniyorsa onlar yazılarak "Parçaları Yarat" tuşu ile disk parçacıkları yaratılır.

Eğer FTP seçeneği işaretlenirse Windows2000 ortamında doküman sunucu üzerinde FTP sunucudada bir site yaratılacaktır.

7.22 SMS Seçeneklerinin Belirlenmesi

SMS sunucu seçeneklerinin belirleneceği paneldir. SMS Sunucu bağlandığı kütüphane

sunucusunun bu değerlerin yazılacağı DMSSMSOPTIONS adlı tablodan okuyarak açılır ve bu parametrelere göre çalışır.



Şekil 7.11 SMS seçeneklerinin belirtilmesi

8. PERFORMANS

Doküman yönetim sistemleri iki amaca hizmet edebilirler. Bunlardan bir tanesi arşiv için kullanılmak, diğeri ise iş akışına yardımcı olmaktır.

Özellikle iş akışı trafiğinin çok yoğun olduğu firmalarda aynı anda bir çok kişi tarafından listelenen çalışma listelerinin tasarımı performans açısından çok önemlidir.

Bununla beraber arama yapma olayı kütüphane veritabanını etkileyebilecektir. Özellikle arama yapılabilir alanlarda performans artırıcı algoritmaların gerçekleştirilmesi gerekmektedir.

Performansı etkileyebilecek bir diğer etken de klasörlerin listelenmesidir.

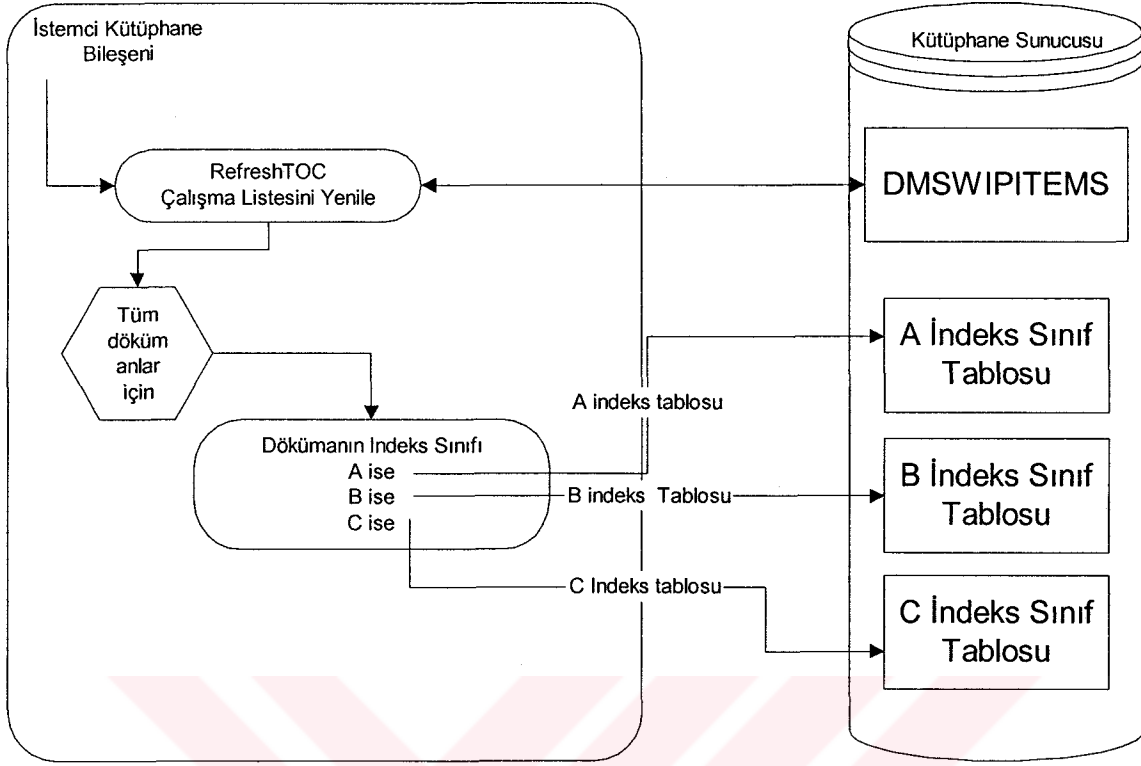
8.1 Çalışma Listelerinin Listelenmesi

Bilindiği gibi çalışma listeleri firma içi iş ve doküman akışını gerçekleştirmek için tasarlanmış iş sepetleridir. Her sepeti bir veya birden fazla kullanıcı görebilir. Bunlar giriş kontrol listeleri sayesinde gerçekleştirilirler.

Çalışma listelerinin listelenmesi özellikle iş akışı yoğun olan bir firma için performansı iyi veya kötü yönde etkilemesi açısından çok önemlidir.

Çalışma listeleri içinde bulunabilecek doküman veya klasörler farklı indeks sınıflarında indekslenebilirler. Bunun anlamı, bir çalışma listesini DMSWIPITEMS tablosunu kullanarak listelediğimizde çalışma listesinin içindeki doküman veya klasörlerin indeks bilgilerine ulaşabilmemiz için kütüphane sunucu veritabanında farklı tablolara gitmek zorunda olduğumuzdur.

Bu durum Şekil 8.1’de verilmiştir. Bir çalışma listesi içinde D1,D2,D3 adında 3 doküman olsun ve bu dokümanların sırayla A,B,C indeks sınıfları içinde indekslemlensin. İstemci bileşeni açısından düşünürsek, yapması gereken iş listeyi yenilemek ve içinde bulunan dokümanların **her biri için** tekrar kütüphane sunucusunda ilgili tabloya gidip bu dokümanların indeks bilgilerini yenilemek olması zorunluluğudur. Bu n doküman içeren bir liste için n+1 defa kütüphane sunucusuna gidilmesi gerektiği demektir.



Şekil 8.1 Çalışma listesi yenileme

Yani çalışma listesi yenilenmek istediğinde Şekil 8.1 de gösterildiği gibi ilk olarak çalışma listesinin kodu ile (WORKBASKET_ID) DMSWIPITEMS tablosuna gidecek ve kendi listesi içinde hangi doküman veya klasörlerin var olduğunu o doküman veya klasörlerin indeks sınıf kodlarıyla beraber okuyacaktır.

Bundan sonra yapması gereken bulduğu D1, D2 ve D3 dokümanlarının indeks sınıflarının tablolarına gidip indeks bilgilerini almaktır. D1 için A indeks sınıf tablosuna, D2 için B indeks sınıf tablosuna, C için ise C indeks tablosuna gidip indeks bilgilerini alır. 3 doküman listelemek için 4 statik prosedür çalıştırmak zorundadır.

Bu bir senaryodur ve bir çok doküman yönetim sisteminde kullanılmaktadır. Mesela IBM Visual Info doküman yönetim sisteminde böyle bir yapı kurulmuştur.

Şimdi bu çalışma şeklinin ne tür kayıpları olduğunu araştıralım. Çünkü bizim tasarladığımız sistem böyle çalışmayacaktır.

Bir çalışma listesine genellikle birden fazla insan tanımlıdır. Bu insanlar listeden bir iş alır, işlerini bitirir tekrar listeyi yeniler ve yeni bir iş alırlar.

Bunun sisteme getirdiği yükü hesaplayalım.

Çalışma listesinde çalışan kişi sayısı K olsun.

Çalışma listesindeki iş sayısı X den başlayıp azalsın.

Kullanıcılar birim zamanda aynı iş yapabilmekteler.

Öyle ise;

Birinci döngüde listeleme $\rightarrow X*K$ sayıda olur ve

K iş bitirilir.

İkinci döngüde iş sayısı $X-K$ ve listelenen sayı $(X-K)*K$ olacaktır.

Üçüncü döngüde iş sayısı $X-2*K$ ve listelenen sayı $(X-2*K)*K$ olacaktır.

Döndü sayısı X / K ile ifade edilebilir.

O zaman formül;

$$\sum_{i=1}^{X/K} (X - (i-1)K)K \quad (8.1)$$

formülü ile hesaplanır. X/K 'nın tamsayı değeri alınır. Bunun sonucunda

$$SP = (X^2 + X*K) / 2 \quad (8.2)$$

formülü ortaya çıkar. SP içinde X adet doküman bulunan ve K kişi tarafından yapılan işin bitirilebilmesi için indeks bilgilerini okumak için çalıştırılması gereken statik prosedür sayısıdır.

Bu gerçekten çalışma listesi sayısının fazla, çalışan kişinin az olduğu ortamlarda korkunç bir sayıdır. Görüldüğü gibi aynı zamanda çalışma listesindeki eleman sayısının karesi ile doğru orantılı bir şekilde artmaktadır.

Burada çözüm formülden görülmektedir. Formül bize çalışma listelerinin mantık çerçevesi içinde bölünebildiği kadar fazla bölünmesi gerektiğini ortaya koyar.

Tasarlanan sistem bu sorunu değişik bir yolla çözmüştür. Yalnız bunu bir olasılık hesabını göz önüne alarak ortaya koyulmuş bir çözümdür.

Çalışma listeleri içinde dokümanların indeks bilgilerinin güncellenme olasılığı çok düşüktür. Çünkü genellikle dokümanlar indeks sınıflarında bir kere indekslendiklerinden sonra kolay kolay bir daha indeks bilgileri değiştirilmez.

Buda bizim XML türü bir çözüm üretmemize neden olmuştur. DMSWIPITEMS tablosuna ITEM_INFO adında bir kolon eklenmiştir. Bu kolon 3500 karakterlik değişebilir uzunlukta bir alana sahiptir. Dokümanın indeks sınıfındaki bilgileri değiştirildiğinde istemci kütüphane bileşeni otomatik olarak dokümanın tüm indeks bilgilerini DMSWIPITEMS tablosunda günceller. Böylece DMSWIPITEMS tablosu içindeki nesnelerin indeks bilgileride beraberinde tutulur. Bu bir çalışma listesinin içindeki bilgilerin **bir seferde** listelenebilmesine olanak sağlar.

Bu gerçekten önemli bir geliştirmedir. Çünkü K kişinin çalıştığı bir çalışma listesinde X doküman varsa indeks bilgilerini almak için hiç bir statik prosedür ekstra çalıştırılmaz.

Yani güncelleme işlemi biraz daha zahmetli olmasına karşın özellikle çalışma listelerinin performansı çok büyük ölçüde geliştirilmiştir.

8.2 Klasörlerin Listelenmesi

Klasörlerin listelenme mantığı neredeyse çalışma listelerinin listelenme mantığıyla aynıdır. Tek fark çalışma listelerinin içerdikleri doküman veya klasörler DMSWIPITEMS tablosundan bulunurken, klasörlerin içerdikleri doküman veya klasörler DMSLINK tablosundan bulunur. Arkasından yapılan indeks bilgilerini yenileme işi birebir çalışma listelerinde yapılanla aynıdır.

Tasarladığımız sistemde biz de buraya dokunmadık. Çünkü güncelleme zamanını iyice kötüye götürmek istemedik. Çünkü klasörler genellikle arşiv amaçlı kullanılan nesnelerdir. Yani çok yoğun bir işleme maruz kalmazlar. Bundan dolayı birim zamanda sisteme getirecekleri yük oldukça azdır. Ayrıca içindeki dokümanlar veya klasörler bilgilerinin güncellenme olasılığında bir hayli fazladır.

8.3 Arama Performansı

Sistemde arama yapılması performansı etkileyebilecek faktörlerden bir tanesidir. Çünkü kütüphane sunucusu merkezidir ve arama süresinin uzun olması tüm sistem performansını etkileyebilir.

Arama genellikle bir indeks sınıfındaki belli dokümanlara veya belli klasörlere yöneliktir. Arama yapmak için kullanıcı standart istemci uygulamasında veya istemci uygulamasını kullanarak geliştirilen başka bir istemci tarafında, arama panelini açar ve arama yapmak istediği indeks sınıfını işaretler. Bundan sonra yapması gereken bu indeks sınıfında varolan kolonlara bazı bilgiler yazıp arama kriterlerini oluşturmak ve arama sonuçlarını daraltmaktır.

Bilindiği gibi indeks sınıfları kütüphane sunucusunda varolan bir veritabanı tablosudur ve doküman veya klasörlerin indeks bilgileri de bu tablonun kolonlarında tutulmaktadır. Bir arama yapıldığında arama kriterlerinde hangi kolonlar üzerinde arama yapılıyorsa veritabanında bu kolonlar üzerinde bir sql cümlesi çalıştırılacaktır. Bu durumda eğer kolonlar üzerinde bir veritabanı indeksi tanımlanmamışsa performans çok kötü bir şekilde etkilenecektir çünkü veritabanı sql'in sonucunu döndürebilmek için tüm tabloyu taraması gerekecektir.

Bundan dolayı sistem yöneticisinin istemci tarafında uygulamayı yazacak kişi ile sürekli irtibat halinde olması gerekmekte ve arama yapılabilecek kolonlar üzerinde veritabanı indeksleri tanımlamalıdır. Aksi takdirde tüm sistem performansı kötü bir şekilde etkilenecektir.

8.4 Sistem Performans Testleri

Sistem performansının ölçülebilebilmesi amacıyla bazı stres testleri yapılmıştır.

Stres testlerinin yapılması için kullanılan makinanın konfigürasyonu aşağıda verilmiştir.

Çizelge 8.1 Performans makina konfigürasyonu

Hafıza	2 x 512 MB
CPU	2 x PIII 500 MHZ
İşletim Sistemi	Windows 2000 Adv.Sunucu
SQL Server	Microsoft SQL2000 sunucu

8.5 Sisteme Doküman Ekleme Performansı

Sisteme belli bir sayıda doküman eklemek ve başarı oranını ve ortalama doküman ekleme zamanını ölçmek için yapılmıştır.

3500 tane 15K ile 100K'dan oluşan TIF doküman arasından iki program rastgele TIF bir dokümanı seçerek sisteme kaydetmeye çalışmıştır. Her bir program 10000 tane TIF doküman atacak ve duracaktır.

Programlardan bir tanesi doküman sunucu üzerinde çalışacak bir diğeri ise uzak bir makinadan doküman ekleyecektir.

Aşağıdaki çizelgede süre ve oranların durumu görülmektedir.

Çizelge 8.2 Sistem doküman yaratma performansı

Kriter	Doküman sunucudan çalışan	Uzaktan çalışan
Ortalama doküman yaratma süresi	351ms	460ms
Başarı oranı	%100	%100
Hata sayısı	0	0
Toplam süre	3510sn	4600sn

10000 doküman eklemesinde hiç bir sorun olmamış, ortalama doküman yaratma süresi doküman sunucu üstünde çalışan program için 351ms, uzaktan çalışan program için 460ms olmuştur.

8.6 Genel Sistem Performansı

Bu stres testi tüm sistemin performansını ölçmek için tasarlanmış genel bir stres testidir. Yazılan ayrı bir programla bazı önemli fonksiyonlar ihtimaller dikkate alınarak binlerce kez çağırılmış ve saatlik performans gözönüne alınmıştır.

Stres programlarından bir tanesi 1sn içinde ortalama 10-15 istekle çalışmaktadırlar. Bunlardan sistemin en son cevap verebilir şekilde çalışabilir durumda olduğu nokta belirlenerek 5 tane açılmış ve saniyede 50-70 istek yaratılmıştır.

Sistemin en önemli fonksiyonlarının ortalama süreleri ve stres uygulaması tarafından seçilme olasılıkları Çizelge 8.3'de verilmiştir.

Çizelge 8.3 Önemli fonksiyonların ortalama gerçekleştirilme süreleri

Fonksiyon	Seçilme Olasılığı	Ortalama Süre (ms)
ÇL Yenileme(100 Dök.)	%20	345
ÇL'sine Yönlendirme	%10	831
Döküman Açma	%10	234
İndeks Bilgilerini Okuma	%10	31
ÇL bilgisini okuma	%10	123
Check Out	%10	125
Check In	%10	75
ÇL'deki en öncelikli dökümanı alma	%20	344

Sistemin 1 saat içinde çalıştırdığı toplam statik prosedür sayısı 214,351 ve dinamik prosedür sayısı 40,400 civarındadır. Bu şu anda orta ölçekli bir bankanın doküman yönetim sistemi gücüne eşdeğer bir güçtür.

9. SONUÇ

Dokümanların fiziksel olarak saklanması, yönetilmesi ve süreç içinde yürütülmesi masraflı, zahmetli ve iş kaybına yol açan bir yöntemdir. Bütün sorunların çözülebilmesi için dokümanların elektronik ortama alınıp burada yönetilebilmesi fikri ortaya atılmış ve doküman yönetim sistemleri tasarlanmaya başlanmıştır.

Dokümanlar elektronik değil de fiziksel olarak yönetildikleri zaman gerçekten üstesinden gelmesi çok zor problemler ortaya çıkar. Bunların en önemlisi bir dokümanın istendiğinde kolayca bulunup erişilmesidir. Milyonlarca klasör veya doküman içinden birine ulaşmak hiç şüphesiz zaman alan bir iştir. Ayrıca dokümanlar firma içi iş akışına sokulduklarında kaybolma riski taşırlar. Yani bir dokümana hiç ulaşamayabilir.

Diğer bir problem bir doküman veya klasörün aynı anda sadece bir kullanıcının hizmetine verilebilmesidir. Bir klasör fiziksel olarak sadece bir kişinin masasında olabilir. Bu süre içerisinde başkası o klasöre ulaşamayacaktır. Buna çözüm olarak dokümanın kopyasını çıkarmak ise insanların farklı kopyalar üzerinde çalışmaları ve dolayısı ile doküman üzerine alınan notları paylaşamamaları sorununa yol açmaktadır.

Dokümanlar sadece yazılı belgeler şeklinde kağıt üzerinde tutulabilirler. Oysa bir doküman yönetim sistemi ile ses dosyasını bile yönetebilmek mümkün olabilecektir.

Dokümanların güvenliğini sağlamak başlı başına bir problemdir. Çünkü dokümanlar klasörlerin içinde bir masadan başka bir masaya dolaştırılırlarken herkez tarafından görülebilir ve değiştirilebilirler. Bunu engellemek uğrunda uygulanan çözümler doküman güvenliğinin önemli olduğu yerlerde büyük masraflara ve zaman kaybına neden olur.

Dokümanları fiziksel olarak saklamak büyük bir masraf gerektirir çünkü işin içine yüzbinlerce doküman, binlerce klasör ve yüzlerce dolap girebilir. Ayrıca bunları alacak büyük bir alana ihtiyaç vardır.

Yukarıda sayılmaya çalışılan ve diğer bir çok problemi çözüme kavuşturmak için doküman yönetim sistemleri geliştirilmiştir. Doküman yönetim sistemleri her tür doküman tipini saklamak, yürütmek ve yönetmek için tasarlanmış sistemlerdir.

Dokümanların elektronik ortamda tutulduğu bir işyerinde kağıt ve yer masrafı çok önemli bir oranda azalır. Dokümanlar ve klasörler aynı anda yüzlerce kişinin hizmetine sunulur. Bir dokümana veya klasöre ulaşılması için kısa bir süre yeterli olacaktır.

Dokümanlar ve klasörlerin güvenliği de istenen seviyede sağlanabilecektir. Örneğin, bir kişi bir dokümanı bulabilir, ancak okuyamaz, veya okuyabilir ancak güncelleyemez. Bu güvenlik sistemini fiziksel doküman sistemlerinde başarmanın imkanı yoktur.

Tezde tasarlanan doküman yönetim sistemlerinde tüm bu problemlere bir çözüm getirilmiştir. Dokümanların saklanması ve yönetilebilmesi için doküman sunucu ve kütüphane sunucusu fikri ortaya atılmıştır. Aynı zamanda klasör mantığı büyük ölçüde geliştirilmiştir.

Fiziksel sistemlerde sadece dokümanlar değil, klasörlerde indeks sınıfı mantığı getirilerek veritabanı tablolarında indeks bilgileriyle tutulmuşlardır. Ayrıca dokümanların ve klasörlerin farklı indeks bilgileriyle indekslenebilmesi sistem tarafından mümkün hale getirilmiştir.

Dokümanların bir iş akışında yürütülebilmesi için çalışma listeleri planlanmış ve tüm doküman tiplerinin bu çalışma listelerinde yürütülmesi başarılmıştır.

Güvenlik en üst düzeyde sağlanmıştır ve çok esnek bir yapısı vardır. Güvenlik için ayrıcalık ve giriş kontrol listeleri mantığı geliştirilmiştir. Ayrıcalıklar en küçük güvenlik birimidir. Mesela bir indeks sınıfında indekslenen dokümanın okunması, güncellenmesi, silinmesi ayrı ayrı ayrıcalık örnekleridir. Bunların hepsi farklı bir ayrıcalıktır. Ayrıcalıklar bir araya getirilerek ayrıcalık setleri oluşturulmuştur. Bu ayrıcalık setleri ile kullanıcılar bir araya getirilerek giriş kontrol listeleri oluşturulmuştur. Yani bir giriş kontrol listesinin içindeki bir kayıt bir kullanıcının o kontrol listesinde hangi ayrıcalıklara sahip olduğunu gösterir. Sistemde bir indeks sınıfı veya bir çalışma listesi yaratılırken bir giriş kontrol listesi bu indeks sınıfına veya çalışma listesine bağlanmalıdır. Yani indeks sınıflarının veya çalışma listelerinin mutlaka bir giriş kontrol listesine bağlanması gerekmektedir. Sistem bunu zorunlu kılar. Böylece kullanıcılar tüm sistemdeki indeks sınıfları ve çalışma listeleri için giriş kontrol listelerinde tanımlandıkları yetki ile yetkilendirilirler. Bu sayede hem indeks sınıflarına hem de çalışma listelerine kullanıcıların ulaşmaları belli ayrıcalıklarla kısıtlanmıştır.

Yedekleme fiziksel bir doküman sistemi için çok pahalı bir işlemdir. Tüm dokümanların ve klasörlerin kopyasının oluşturulması ve bunların başka yerlerde saklanması gerekmektedir. Aynı zamanda her yapılan güncellemede bu işlemin yedeklenen fiziksel sistemde uygulanması gerekir ki işlenen dokümanların disk üzerinde kapladığı alan büyüdüğünde bu başarılması imkan dahilinde olmayan bir meseledir. Tasarlanan elektronik doküman yönetim sistemi bu sorununda üstesinden gelebilmiştir.

Elektronik doküman yönetim sistemlerinde yedekleme işlemine daha basit çözümler getirilmiştir. Tüm dokümanların tutulduğu doküman sunucuların çok büyük olması muhtemel

sabit disk alanları mantıksal parçalara ayrılarak daha küçük disk parçaları oluşturulmuştur. Aynı zamanda güncelleme bu sabit disk parçaçıklarından sadece bir disk parçası üzerine yapılacaktır. Bu işlem işaretlenen disk parçası dolana kadar devam edecektir. Bu disk parçası dolduğunda yeni bir disk parçası seçilecek ve sistemdeki dokümanlarda yapılan güncellemeler bu disk parçası üzerinde kaydedilmeye başlanacak, eski disk parçasına dokunulmayacaktır. Bu durumda her gece kartuş üzerine sadece o anda sistem tarafında kullanılan disk parçasının yedeklemesinin yapılması yeterli olacak ve yedekleme disk parçası üzerindeki doküman sayısına bağlı olmak üzere 0.5-2 saat zaman alacaktır. Bu disk parçası dolduğu zaman kartuşa son yedekleme yapılarak bu kartuş arşive kaldırılacak ve bu dolan disk alanının yedeklemesi ile bir daha uğraşılmayacaktır. Yeni bir kartuş takılarak sistem tarafından yeni seçilen disk parçasının her gece yedeklenmesine başlanacaktır. Böylece sistem sürekli kendisini yedeklemeyi başarır ve çok büyük bir felakette bile çok küçük doküman kayıpları olur.

Tasarlanan sistemde bir çok şey otomatik hale getirilmiştir. Bunlardan bir tanesi, doküman sunucuların sabit disk parça alanlarının takip edilmesi ve bir disk parçası üzerindeki boş alan belirlenen eşik değerinin altına geldiğinde otomatik olarak başka bir disk parçasının seçilmesidir. Böylece sistem yöneticisinin dokümanların kaydedilecekleri yer ile ilgili arada birde olsa güncelleme yapması gerekmeyecektir.

Ayrıca tüm doküman yönetim sisteminin konfigürasyonu merkezidir. Kullanıcıların tanımlanması, kullanıcı gruplarının oluşturulması, ayrıcalık setlerinin yaratılması, giriş kontrol listelerinin oluşturulması, indeks sınıflarının tanımlanması, çalışma listelerinin yaratılması ve yönetilmesi, yeni doküman sunucuların eklenmesi ve disk parça yönetiminin sağlanması gibi bir çok konfigürasyon parametresi merkezi bir yönetim ile yürütülür hale getirilmiştir.

Elektronik doküman yönetim sisteminin fiziksel doküman yönetimine bir çok üstünlüğü vardır. Fiziksel doküman yönetim sistemlerinin elektronik doküman yönetim sistemlerine ise sözünü edebileceğimiz hiçbir üstünlüğü bulunmamaktadır. Firmaların müşterilerine sunduğu kalite farkını kendi iş akışlarının ve sahip oldukları bilgisayar sistemlerinin hızlarının belirlediği bu şiddetli rekabet ortamında doküman yönetim sistemlerinden uzak kalmaları onların gelecekleri açısından pek iç açıcı gözükmemektedir

KAYNAKLAR

IBM, (2000) , “IBM Content Manager for Multiplatforms”, Böblingen

Leymann, F. ve Roller, D., (2000), “Production Workflow Concepts and Techniques”, Böblingen

Pacheco X. ve Teixeira S., (2000) , “Delphi 5 Developer’s Guide” , Sams Publishing

Soukup R. ve Delaney K., (1999), “Inside Microsoft SQL Server 7.0”, Microsoft Press, Redmond

Williams, M., (1999), “Programming Microsoft Windows 2000”, Microsoft Press, Redmond

İnternet kaynakları

EnterFirması, “http://www.enter.com.tr/web/eiscozumleri/index_dokuman_yonetimi.asp?secim=eiscozumleri04”

Gilheany, S., “Projecting the Cost of The Magnetic Disk StorAge over the Next Ten Years”, <http://archivebuilders.com/whitepapers/22011.doc>

Kets Firması, “Doküman Yönetimi”, “<http://www.kets.com/tr/urunler/dokuman.yonetimi/>”

EKLER

Ek 1 Neler geliştirilebilir?

Ek 2 Akış Diagramları



Ek 1 Neler geliştirilebilir?

Firmaların güzümündeki en önemli sorunlarından biri firma içi iş akışlarının düzgün bir şekilde yürütülememesidir. Bu yüzden firma içi üretimde ve verimde büyük kayıplar yaşanmaktadır.

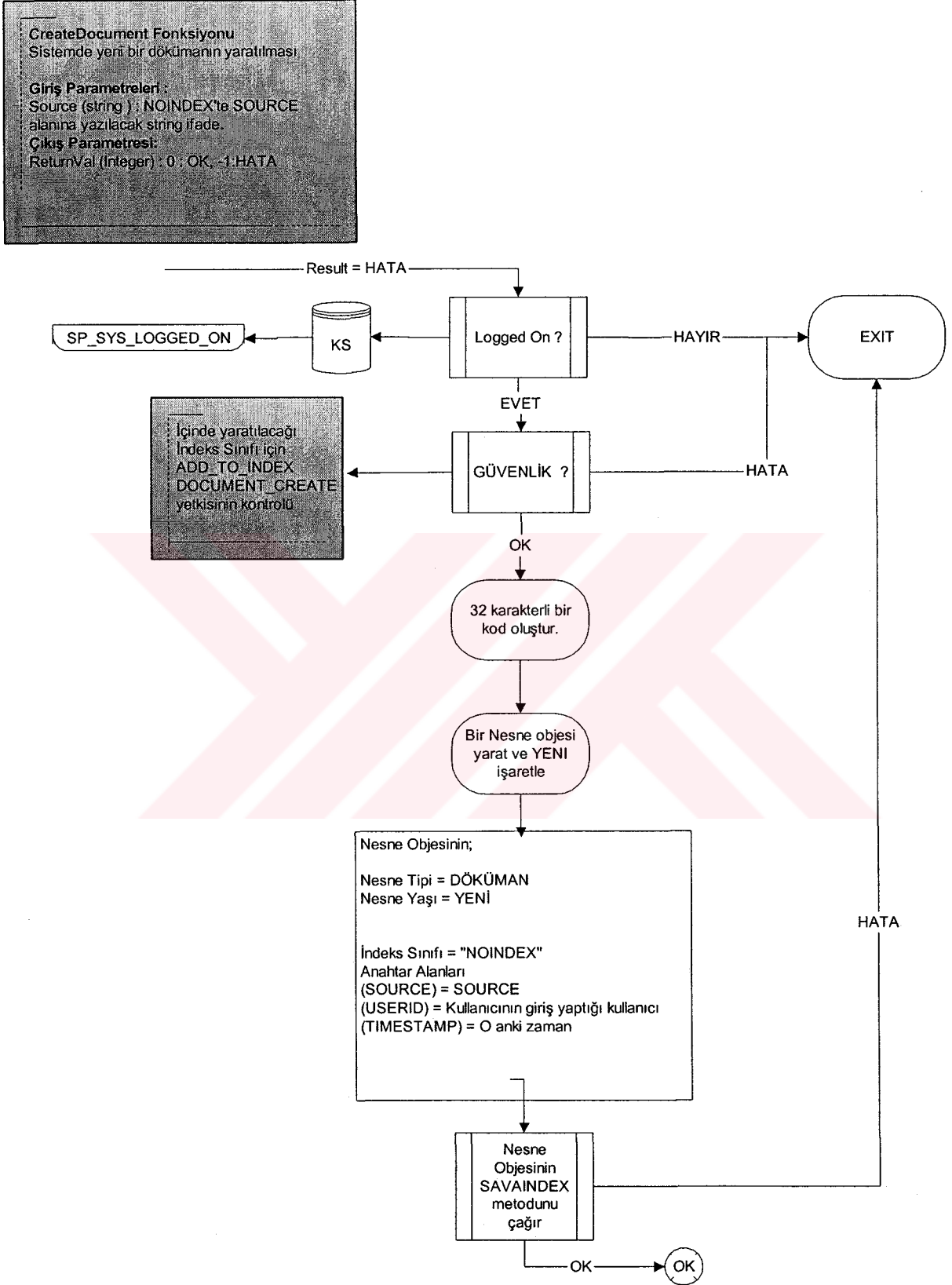
Doküman Yönetim Sisteminin asıl görevi dokümanları kolay ulaşılabilir bir yerde saklamak ve istendiğinde kullanıcının önüne getirmektir. Tabiki çalışma listelerinin kullanılması bir şekilde iş akışını sağlamakta ancak, bu görevi bir anlamda doküman yönetim sisteminin üzerine yazılan uygulamayı kullanan kullanıcıya bırakmaktadır.

Bu nokta geliştirmeye çok açık bir noktadır. Tüm sistemin genelinde prosesler oluşturulması ve bunlara bir iş akışı modelinin verilmesi ve bunları otomatik olarak sistemin yapması çok önemli bir geliştirme olacaktır. Bunun için sistemin yapısına bir iş akışı modeli oturtmak gerekmektedir. Burada düşünülmesi gereken en önemli nokta doküman yönetim sisteminin mi iş akışı sistemine, iş akışı sisteminin mi doküman yönetim sistemine entegre edilmesi gerektiğidir.

Diğer bir geliştirmede kütüphane sunucusunun merkezi olmaktan çıkarılıp dağıtık bir yapıya geçirilmesi ile mümkün olabilir. Böylece sistemin performansı kütüphane sunucusunun işlemci gücüne bağlı olmaktan çıkartılıp dağıtık sistemin sınırsız gücüne eriştirilebilir.

Sistemin geneli yazılacak uygulamalarla intranet ve internete açılabilir. Bunun için doküman yönetim sistemi üzerine bazı internet ve intranet uygulamalarının geliştirilmesi yeterli olacaktır.

Ek 2 Akış Diagramları

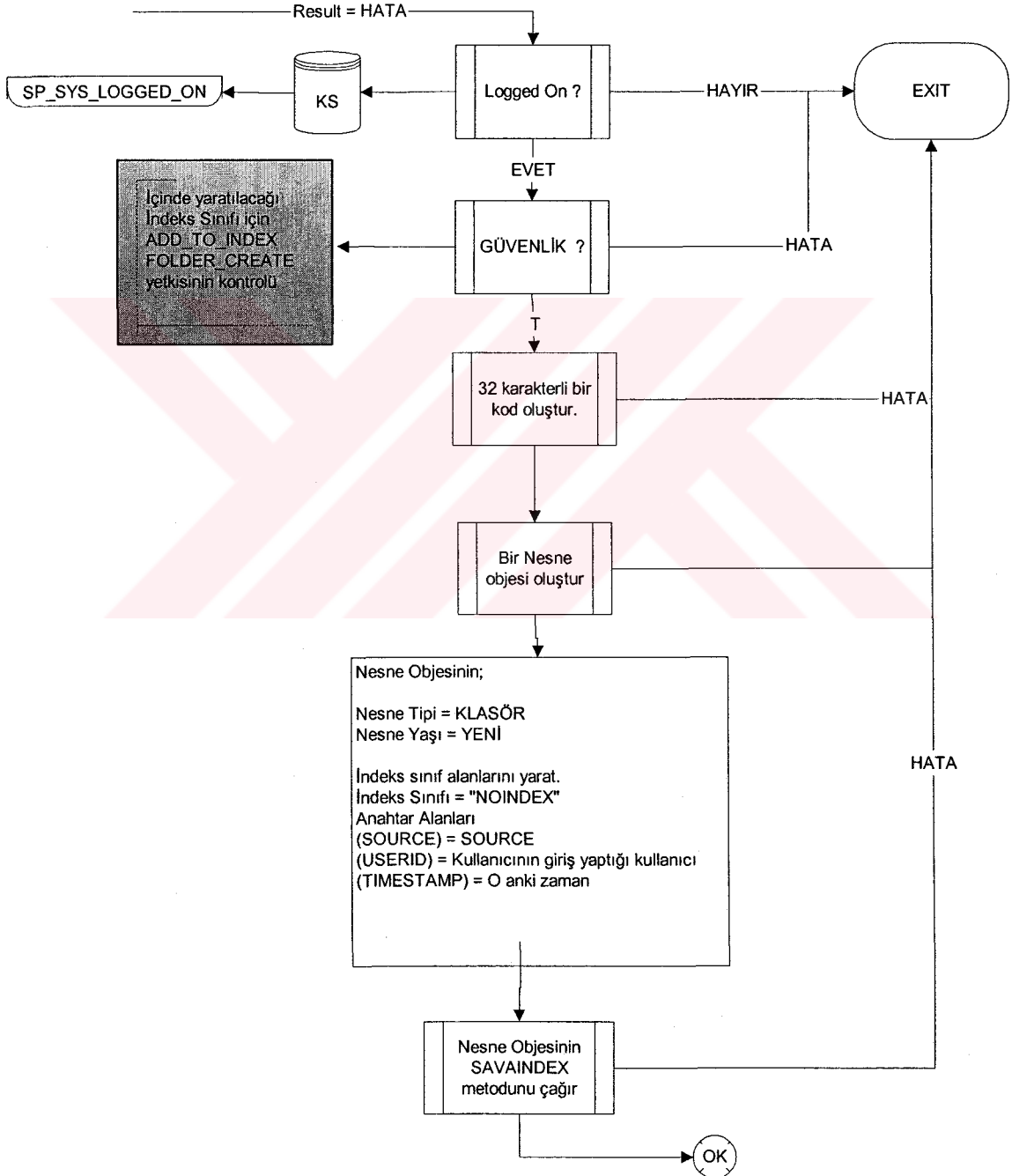


Şekil Ek.2-1 Createdocument metodu akış diagramı

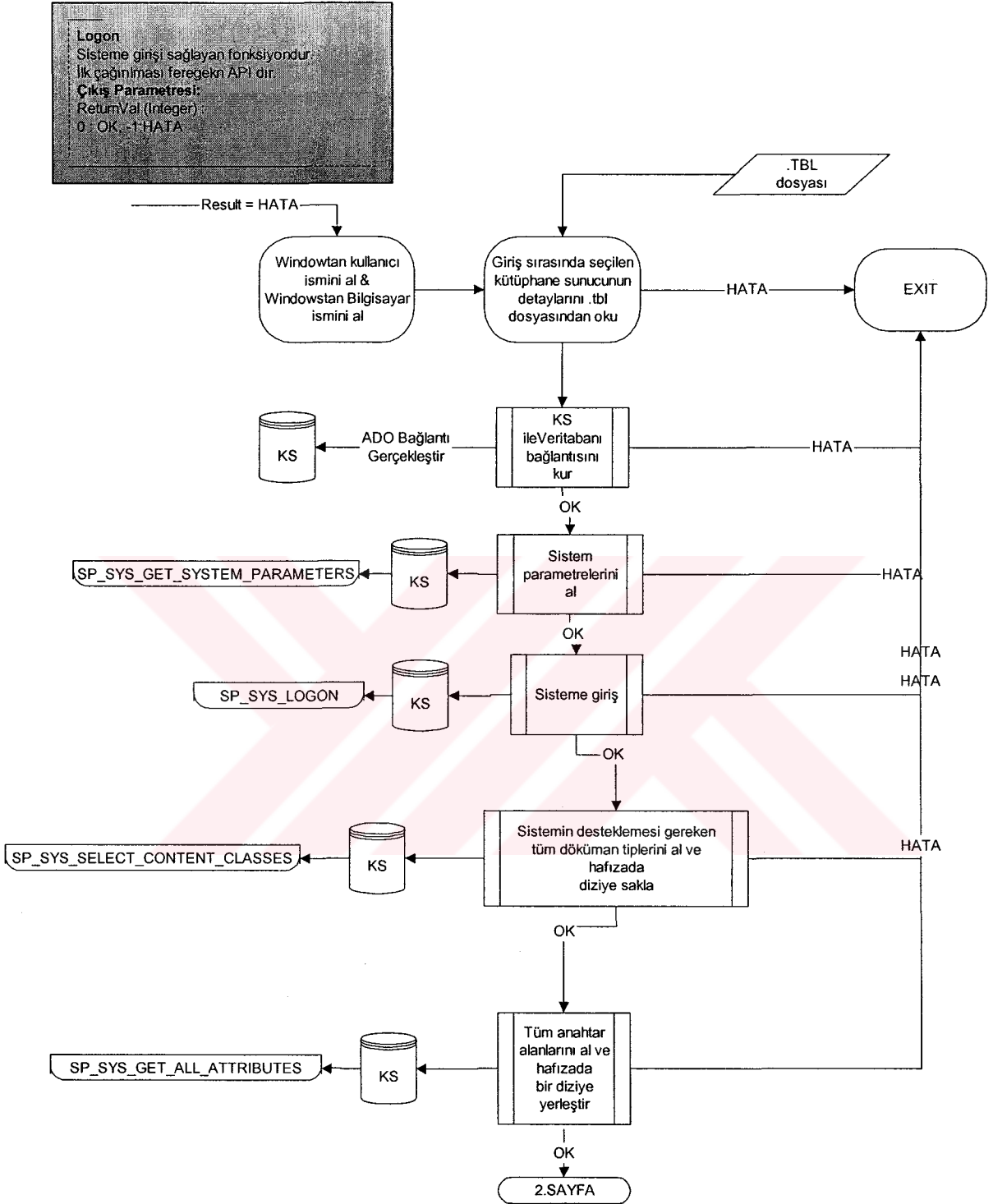
CreateFolder:
Sistemde yeni bir Klasör oluşturma.

Giriş Parametreleri :
Source (string) : NOINDEXte SOURCE alanına yazılacak string ifade.

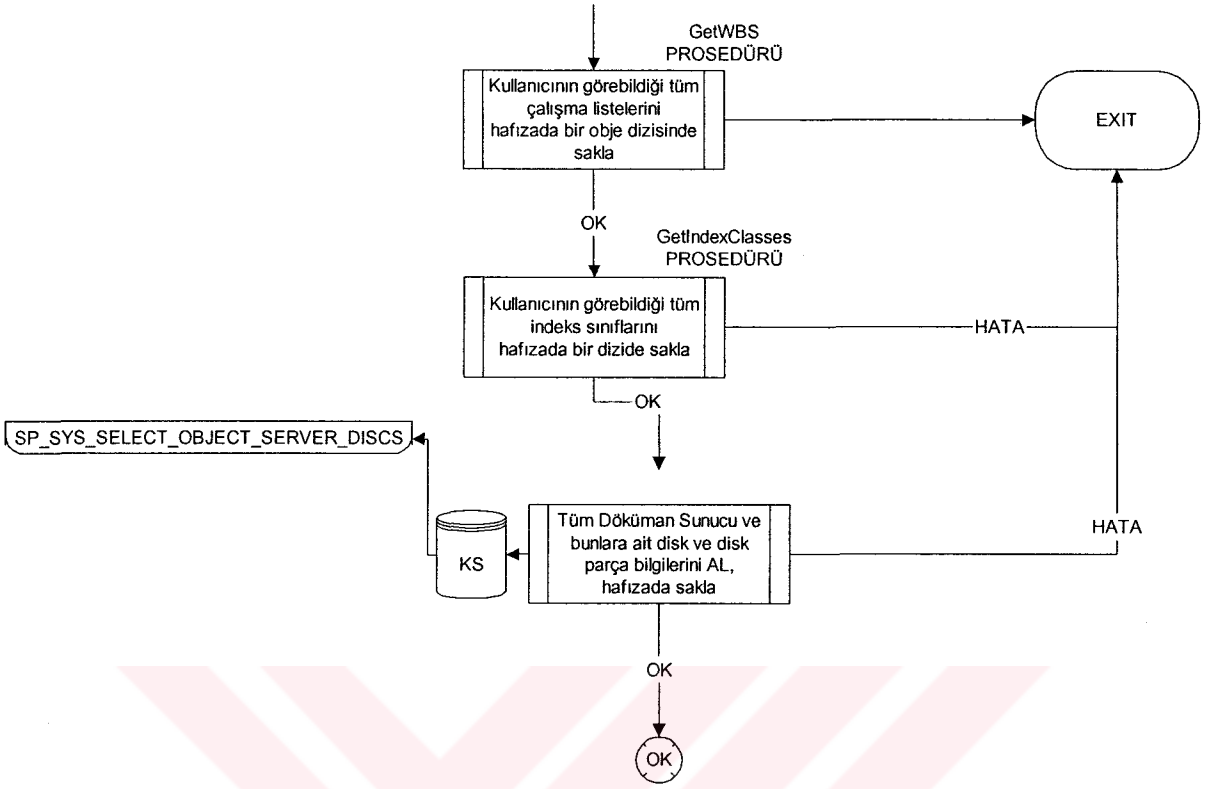
Çıkış Parametresi:
ReturnVal (Integer) : 0 - OK, -1-HATA



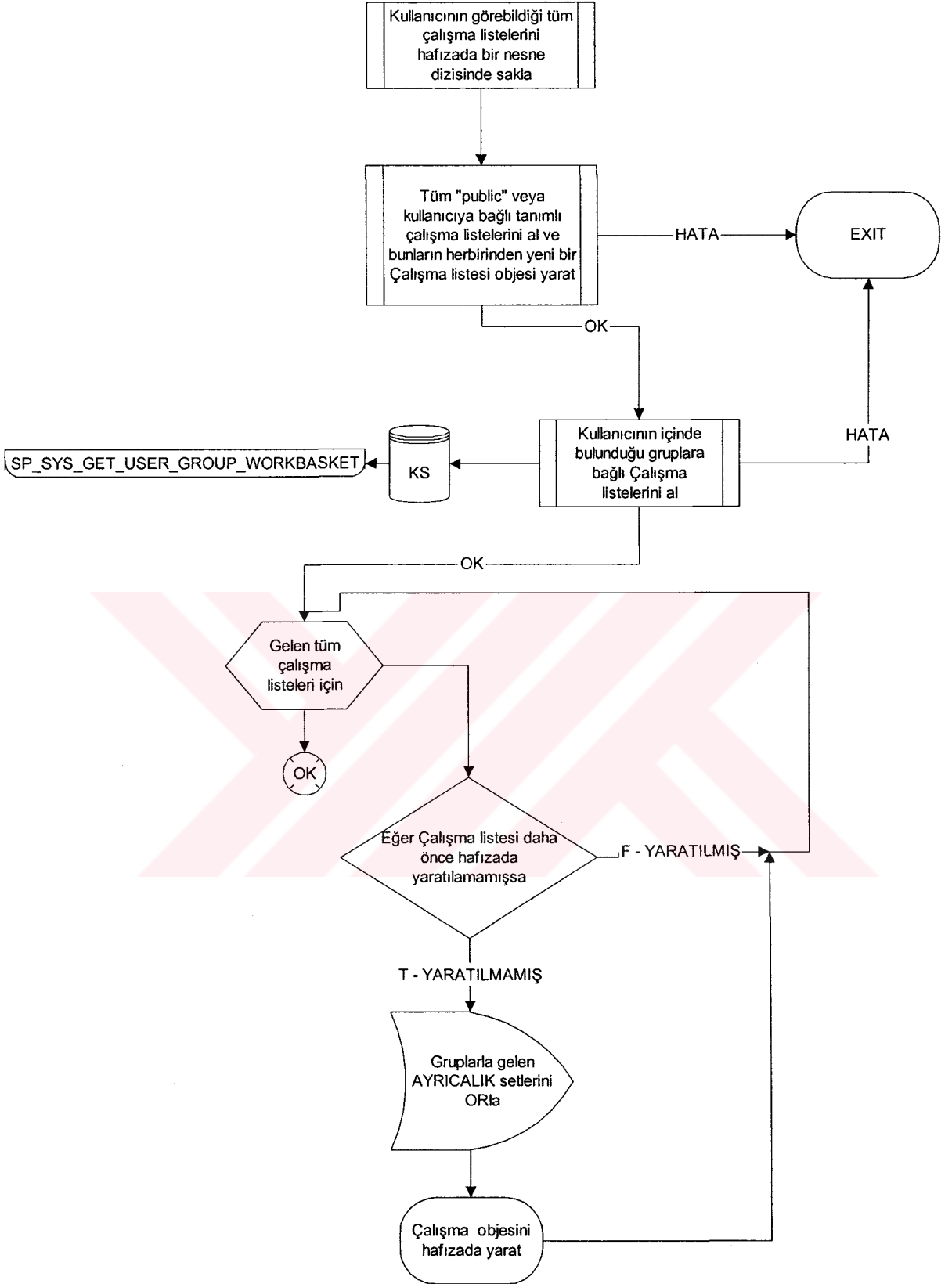
Şekil Ek.2-2 Createfolder metodu akış diagramı



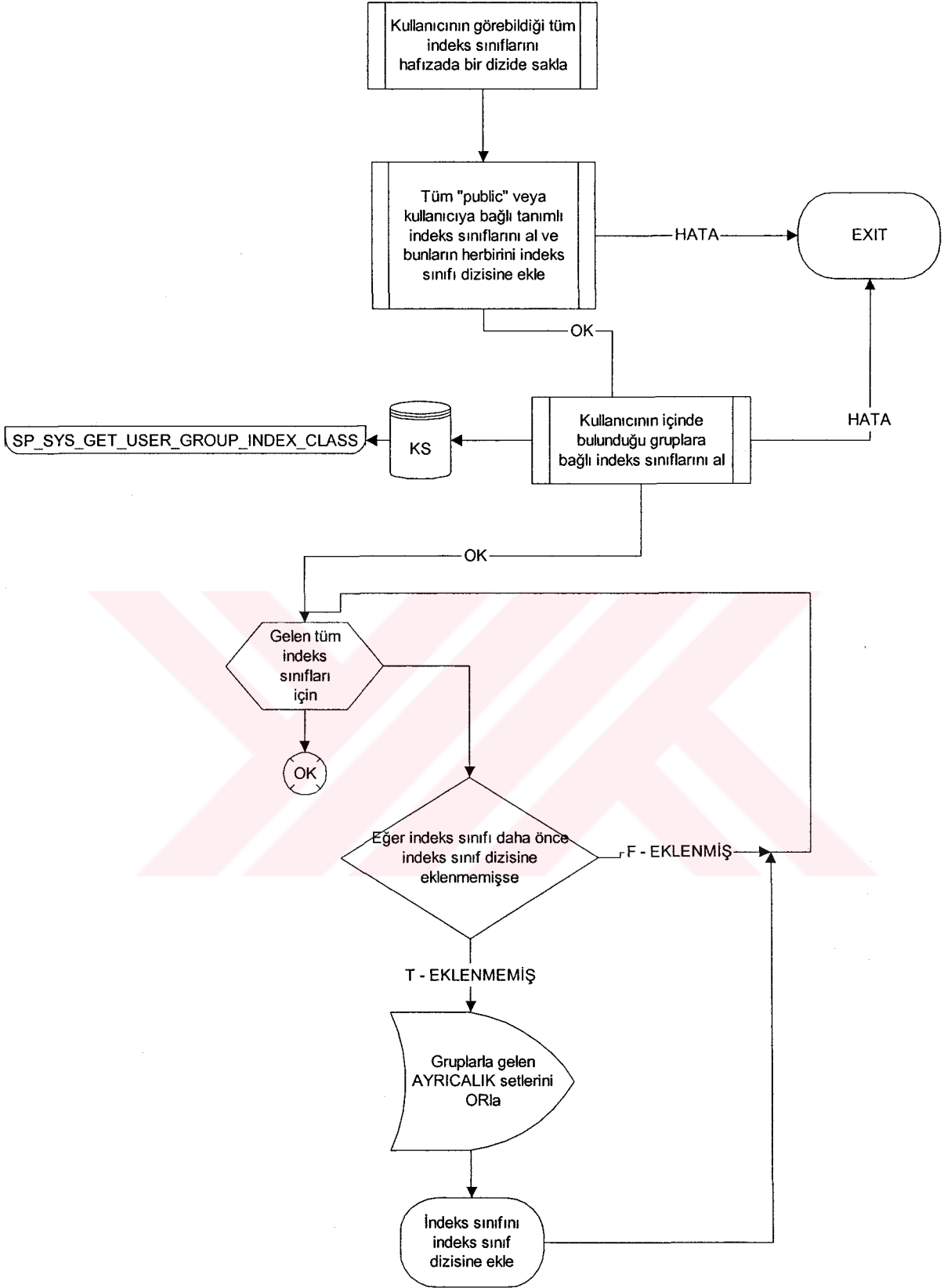
Şekil Ek.2-3 Logon metodu akış diagramı - 1



Şekil Ek.2-4 Logon metodu akış diagramı - 2

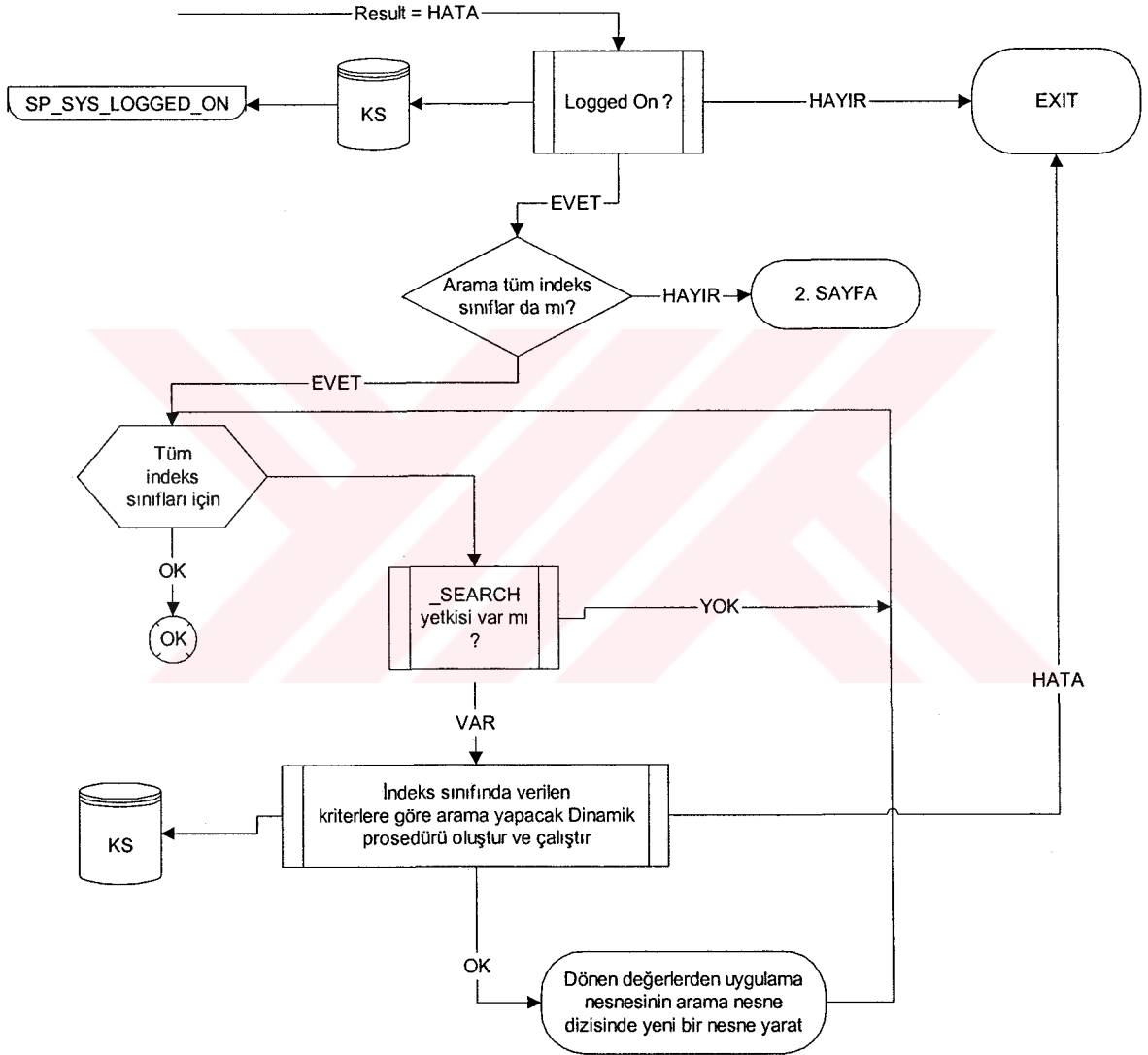


Şekil Ek.2-5 Çalışma listeleri prosedürü akış diagramı

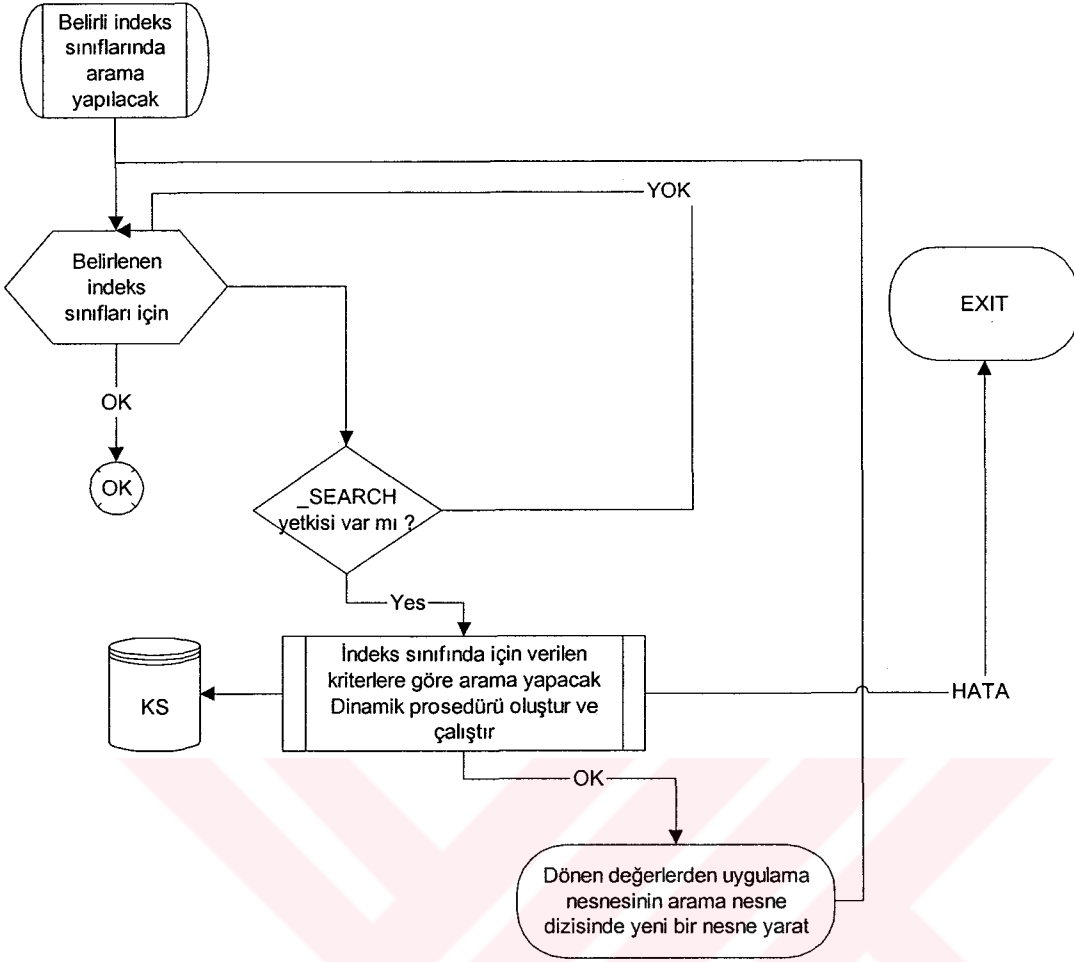


Şekil Ek.2-6 İndeks sınıfları prosedürü akış diagramı

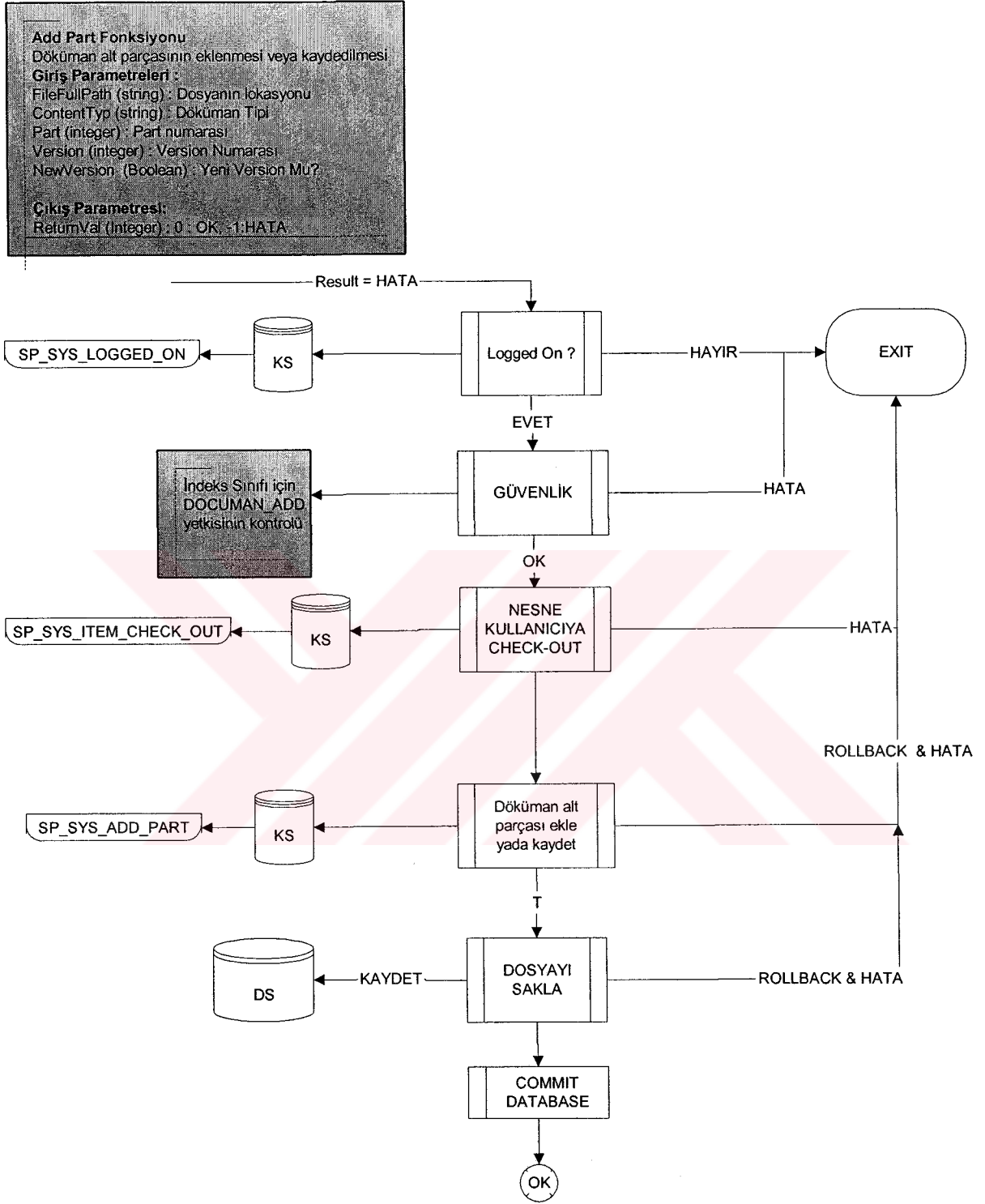
Search Fonksiyonu
Arama yapar
Giriş Parametreleri :
SearchFilter (Integer) :
0 : Sadece Dokümanlar
1 : Sadece klasörler
2 : Hepsini
Çıkış Parametresi:
ReturnVal (Integer) : 0 : OK, -1:HATA



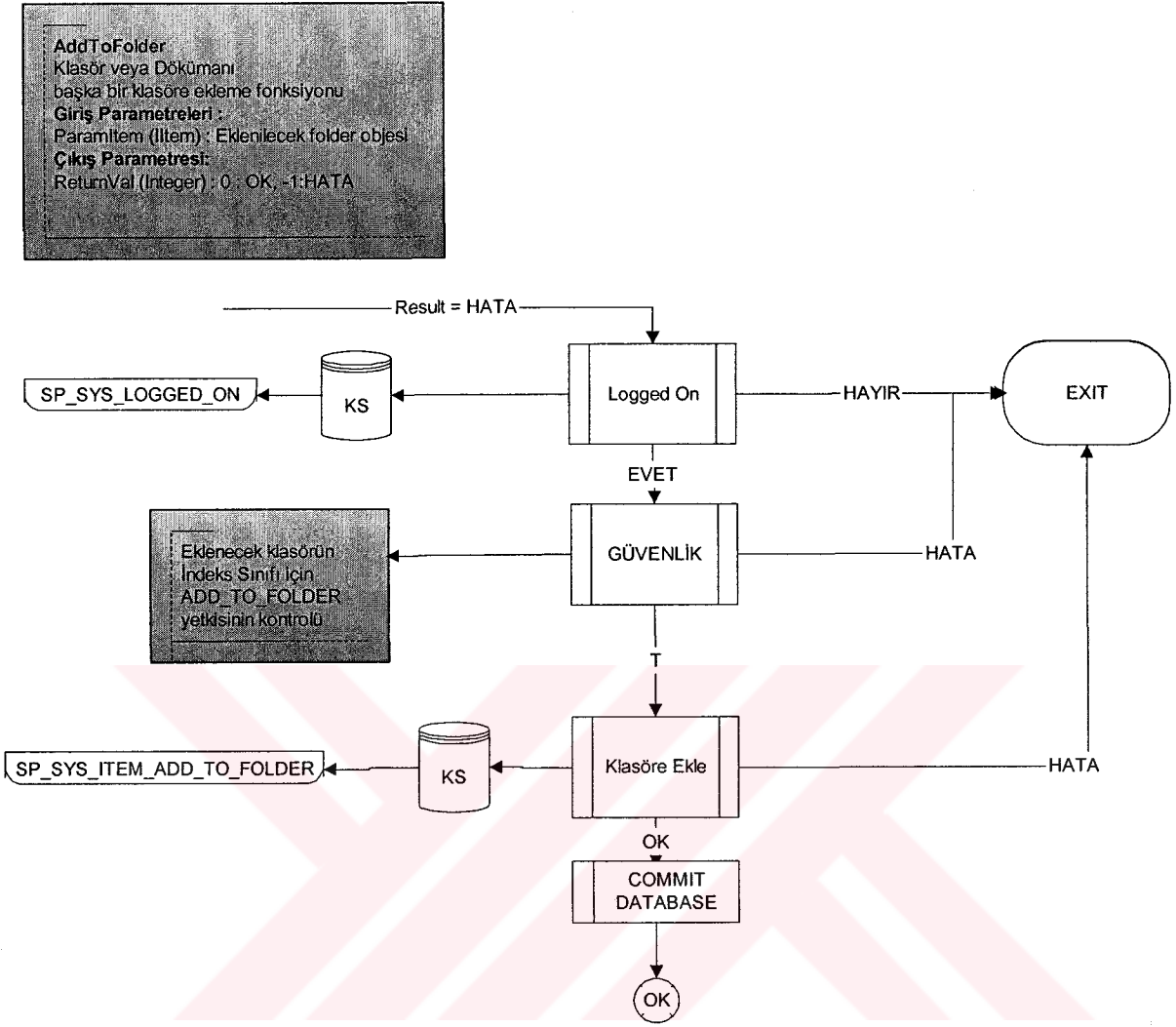
Şekil Ek.2-7 Search metodu akış diagramı - 1



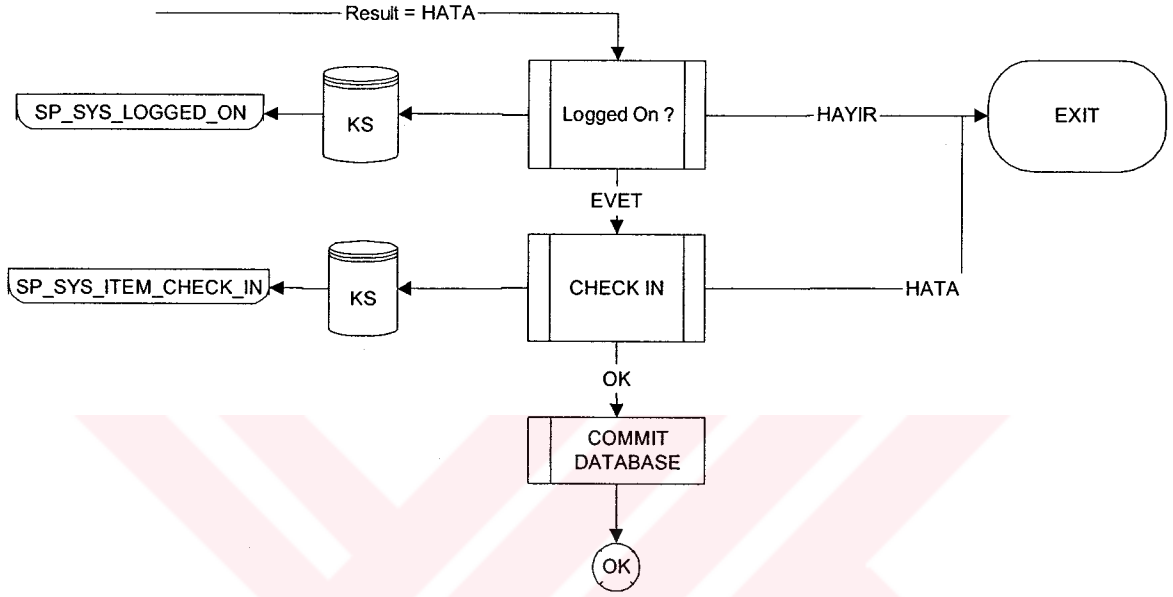
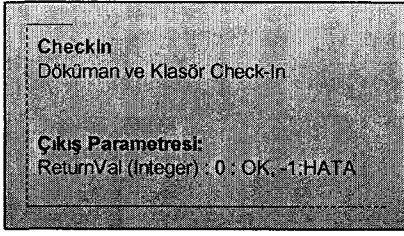
Şekil Ek.2-8 Search metodu akış diagramı - 2



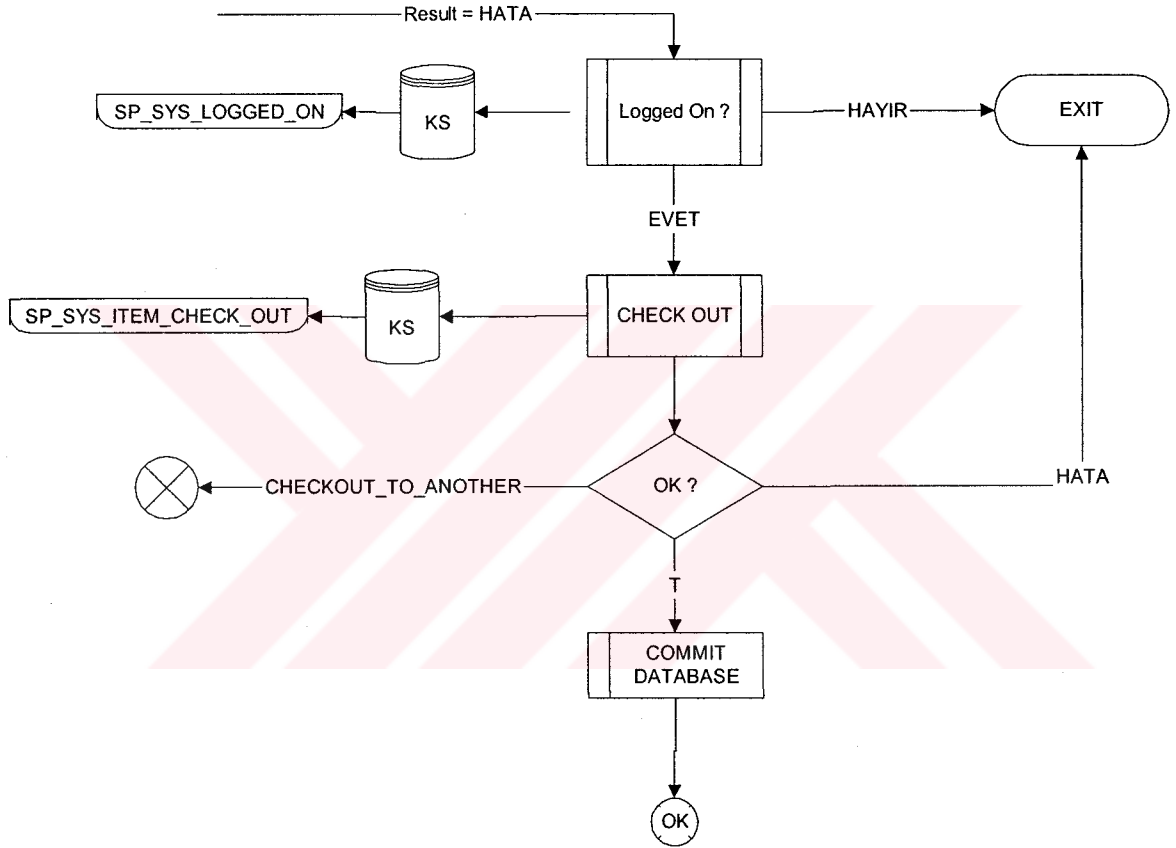
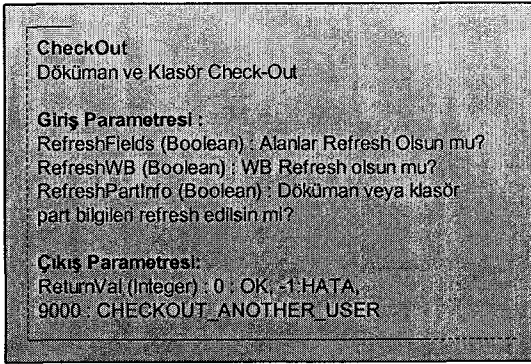
Şekil Ek.2-9 Addpart metodu akış diagramı



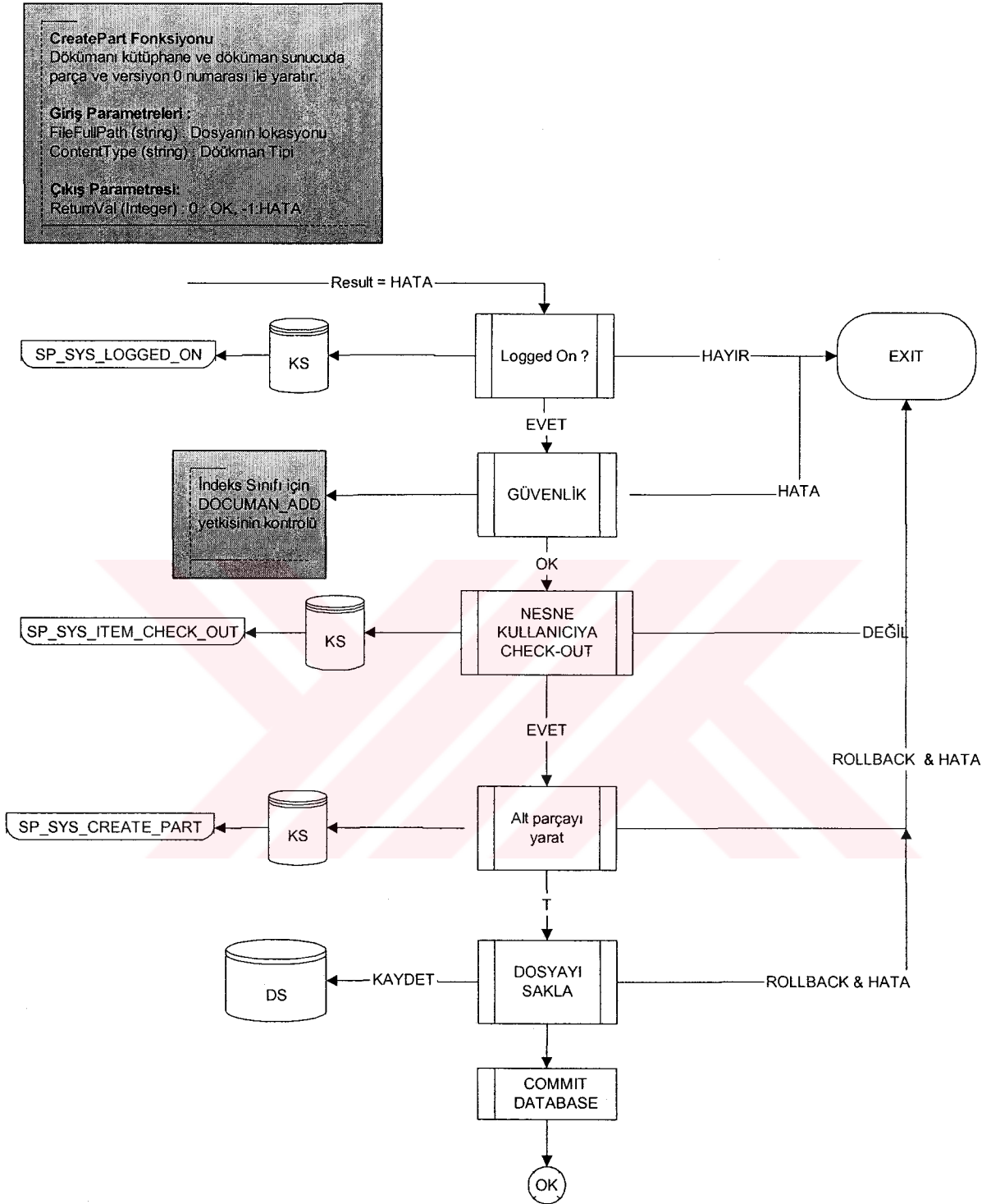
Şekil Ek.2-10 Addtofolder metodu akış diagramı



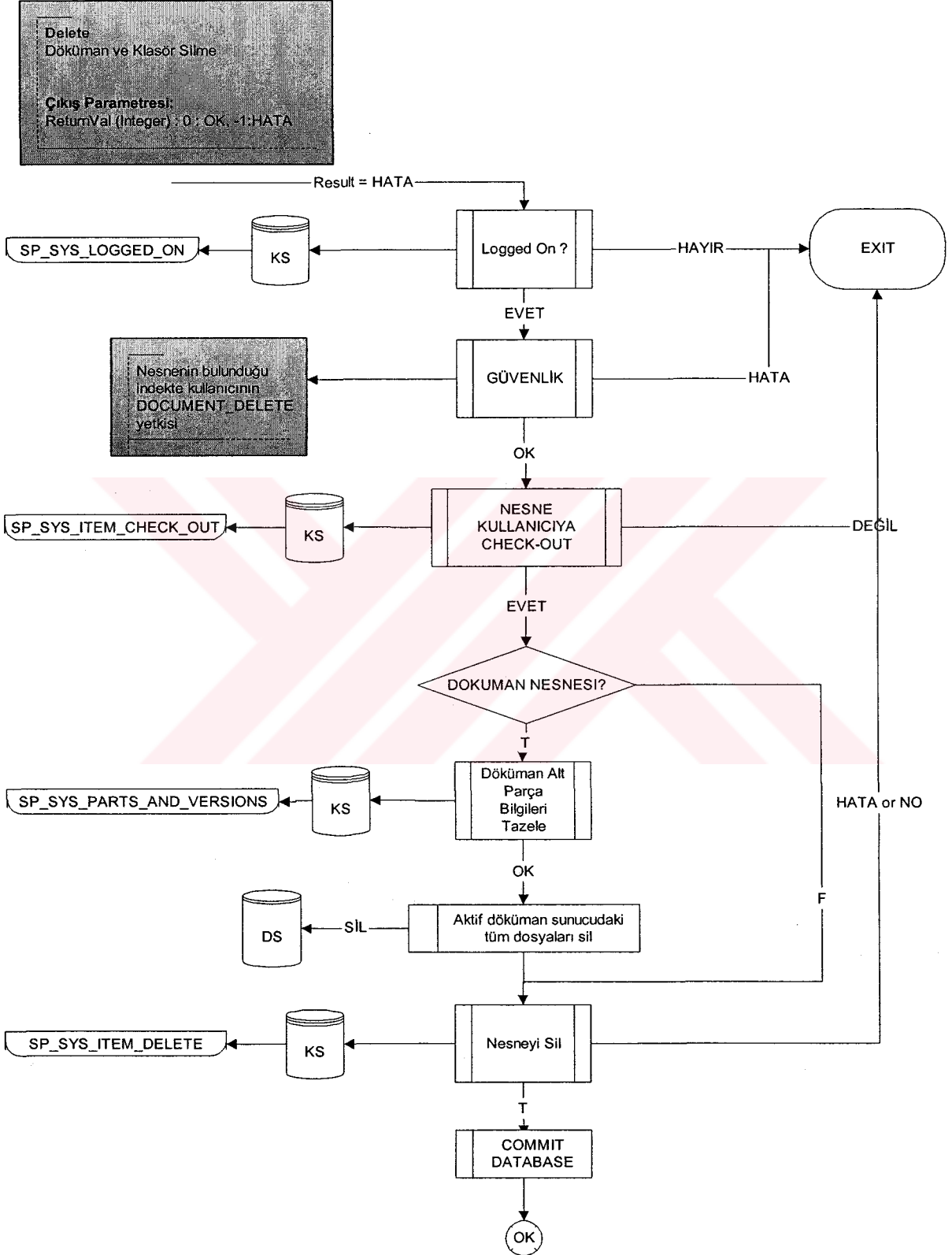
Şekil Ek.2-11 Checkin metodu akış diagramı



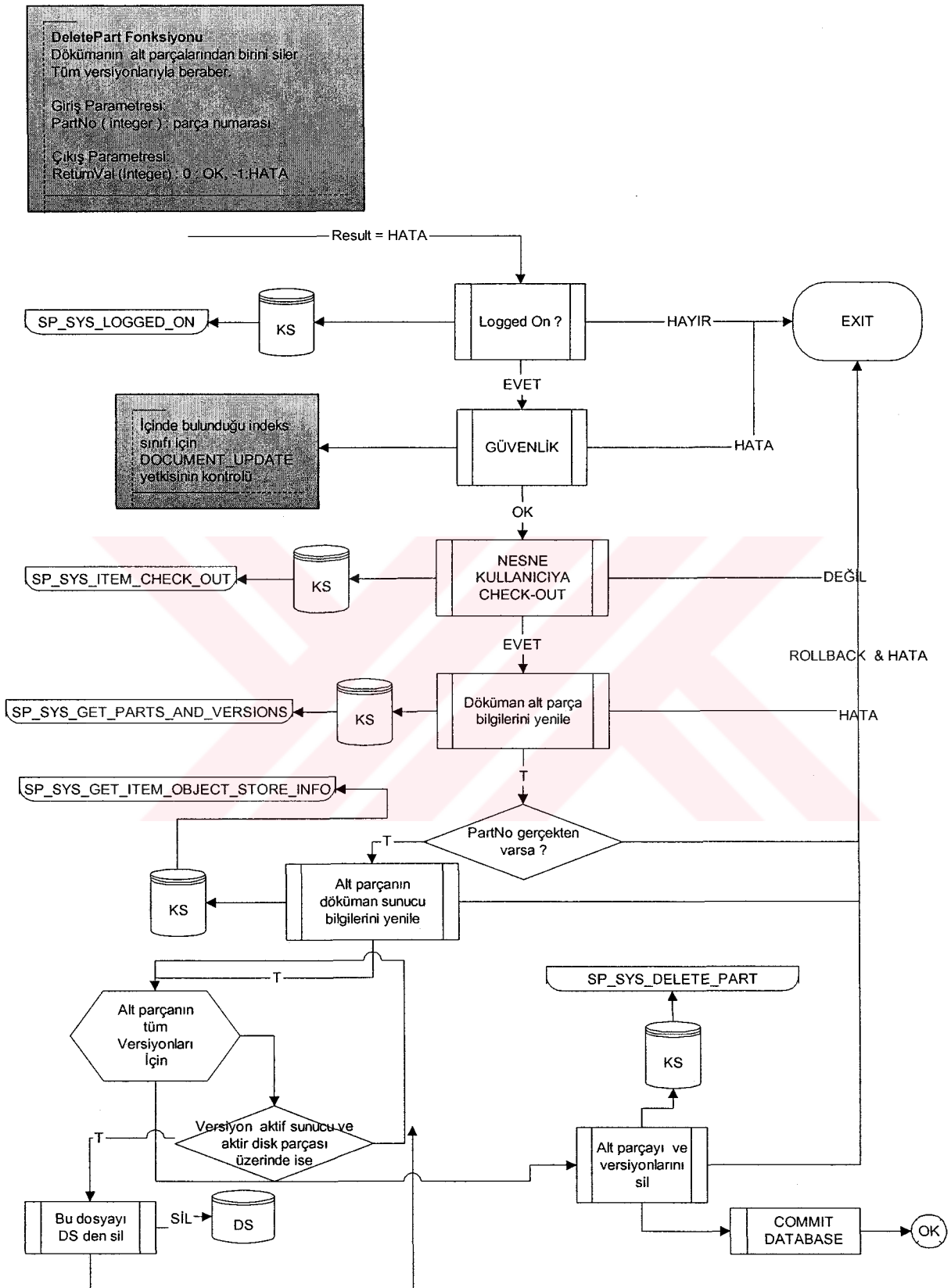
Şekil Ek.2-12 Checkout metodu akış diagramı



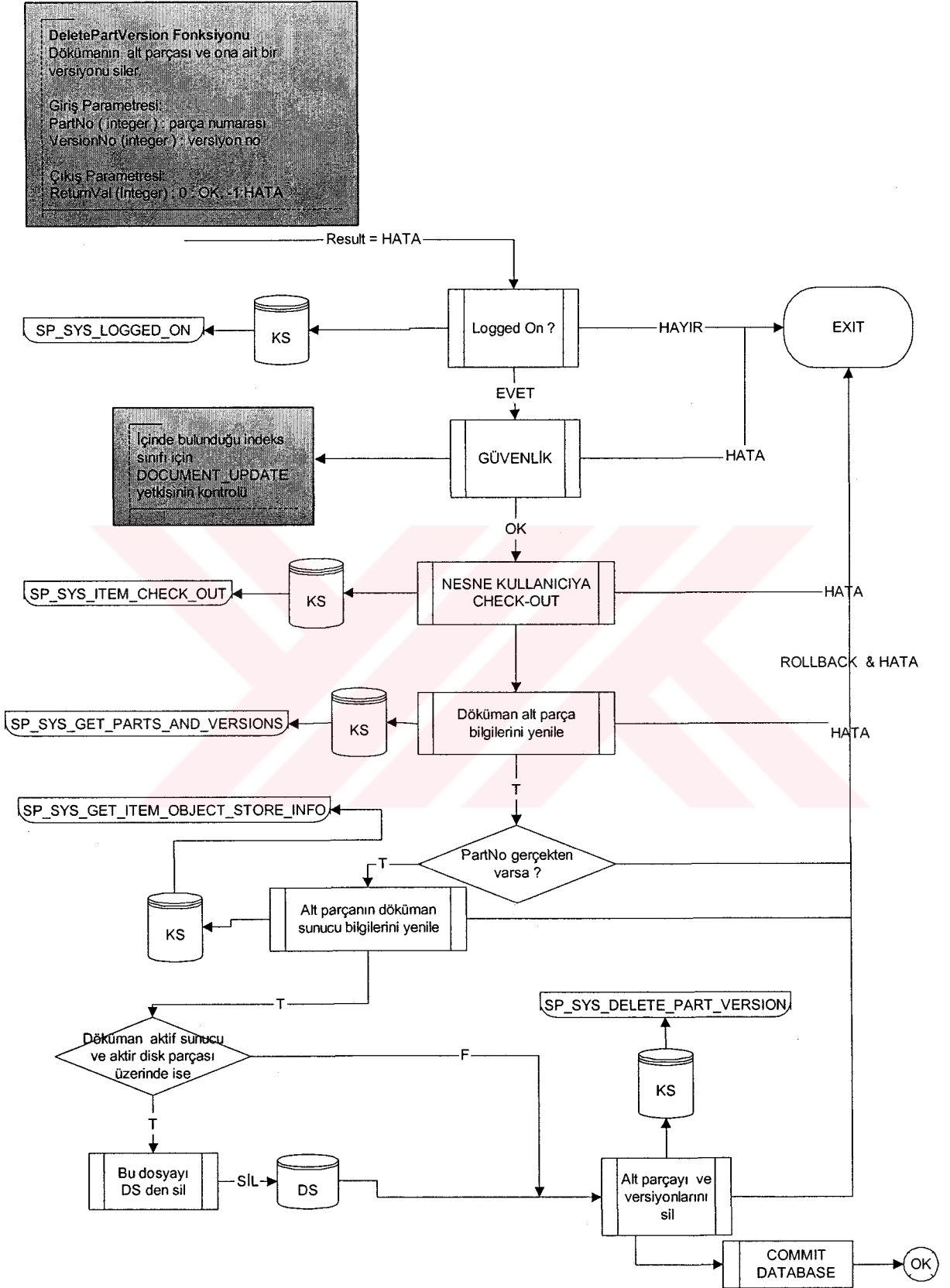
Şekil Ek.2-13 Createpart metodu akış diagramı



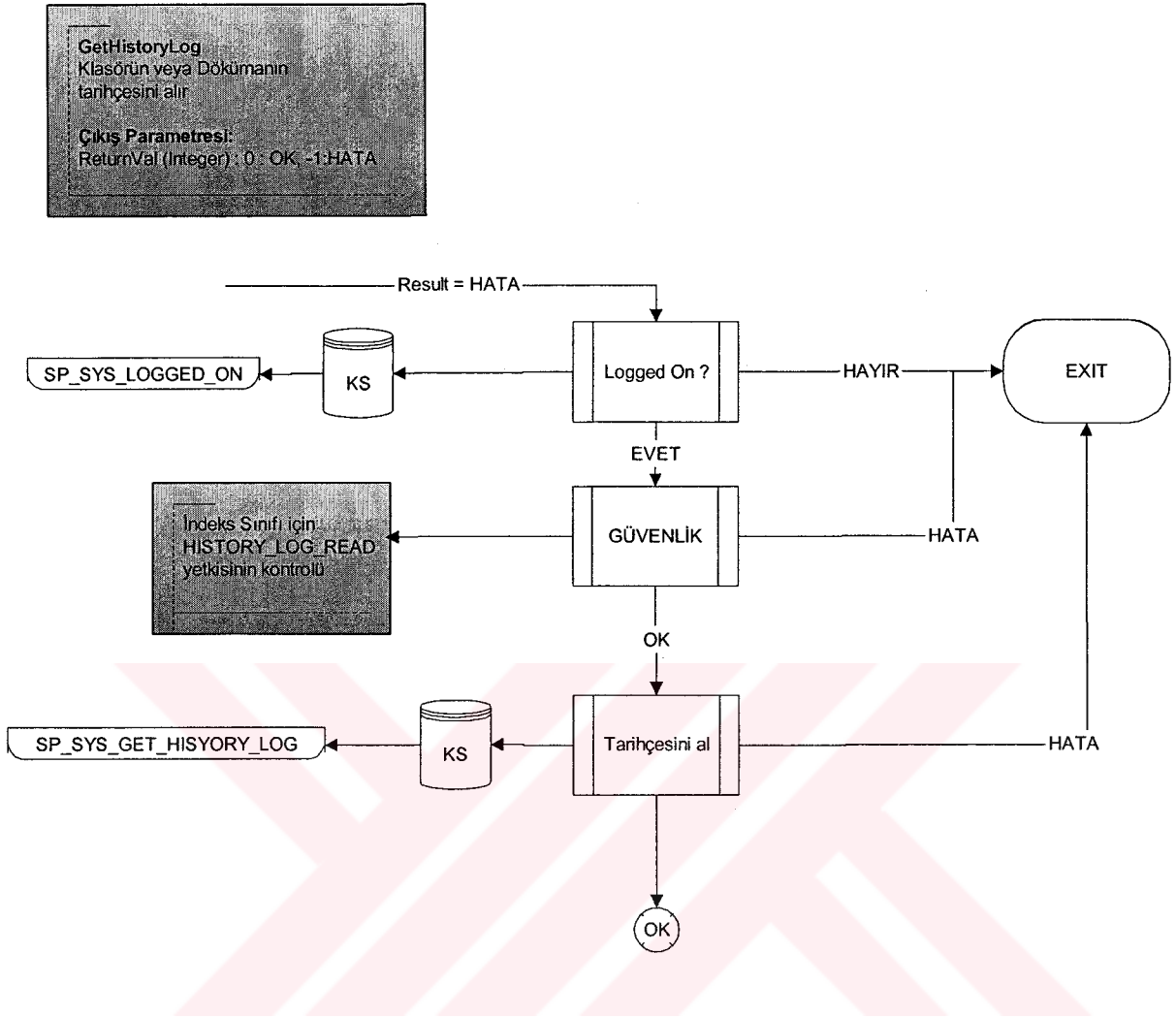
Şekil Ek.2-14 Delete metodu akış diagramı



Şekil Ek.2-15 Deletepart metodu akış diagramı



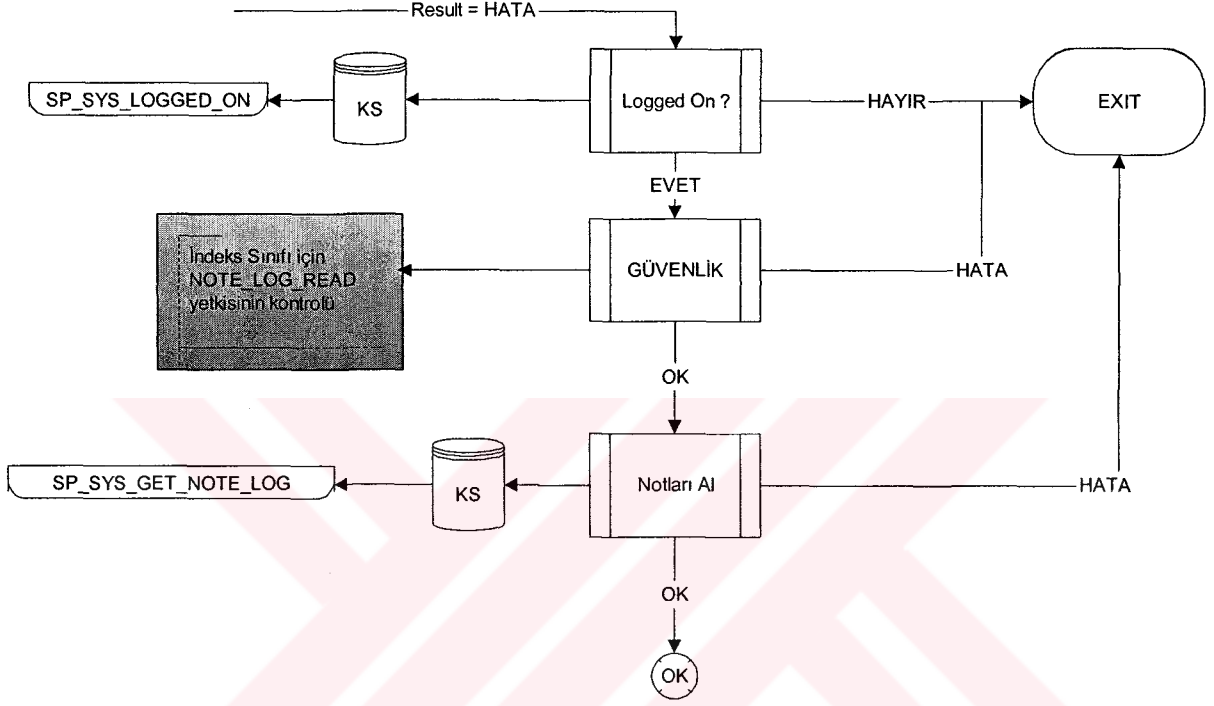
Şekil Ek.2-16 Deletepartversion metodu akış diagramı



Şekil Ek.2-17 Gethistorylog metodu akış diagramı

GetNotes (Klasörün veya Dokümanın saklanan notlarını alır)

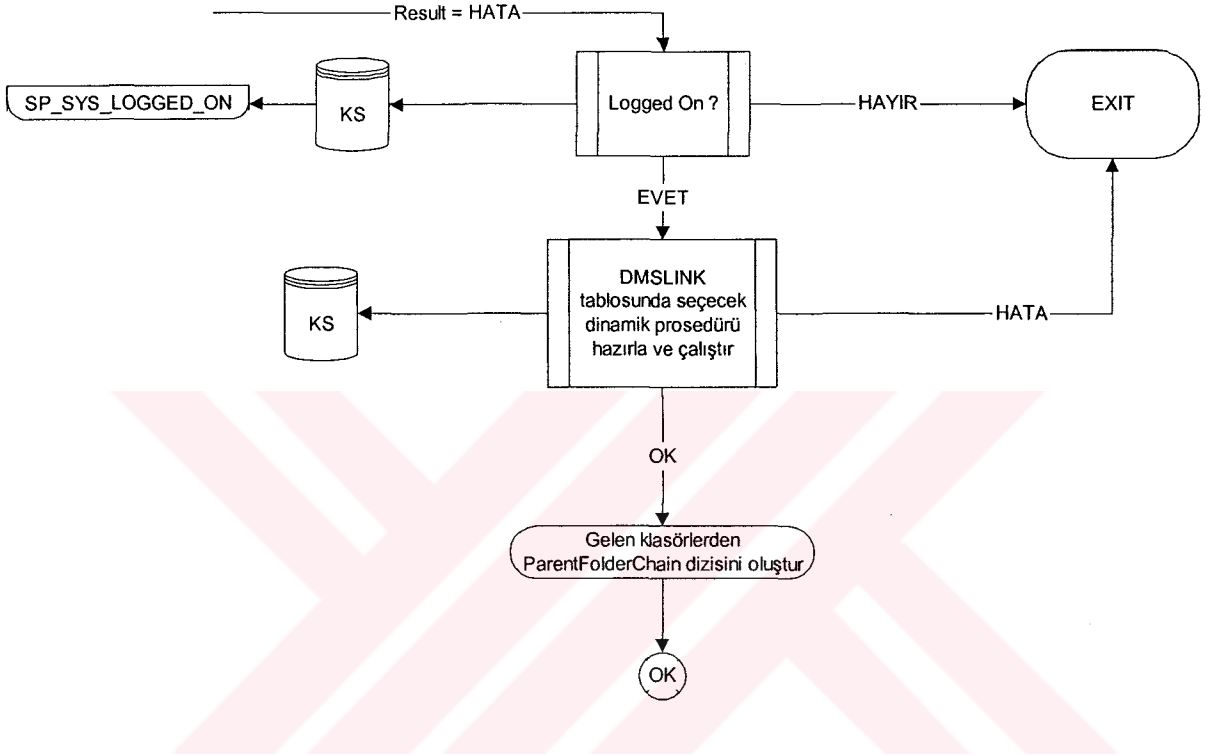
Çıkış Parametresi:
ReturnVal (Integer): 0 - OK, -1:HATA



Şekil Ek.2-18 Getnotes metodu akış diagramı

GetParentFolders (Klasör veya Dökümanın ana klasörlerini bulur)

Çıkış Parametresi:
ReturnVal (Integer) : 0 : OK, -1:HATA

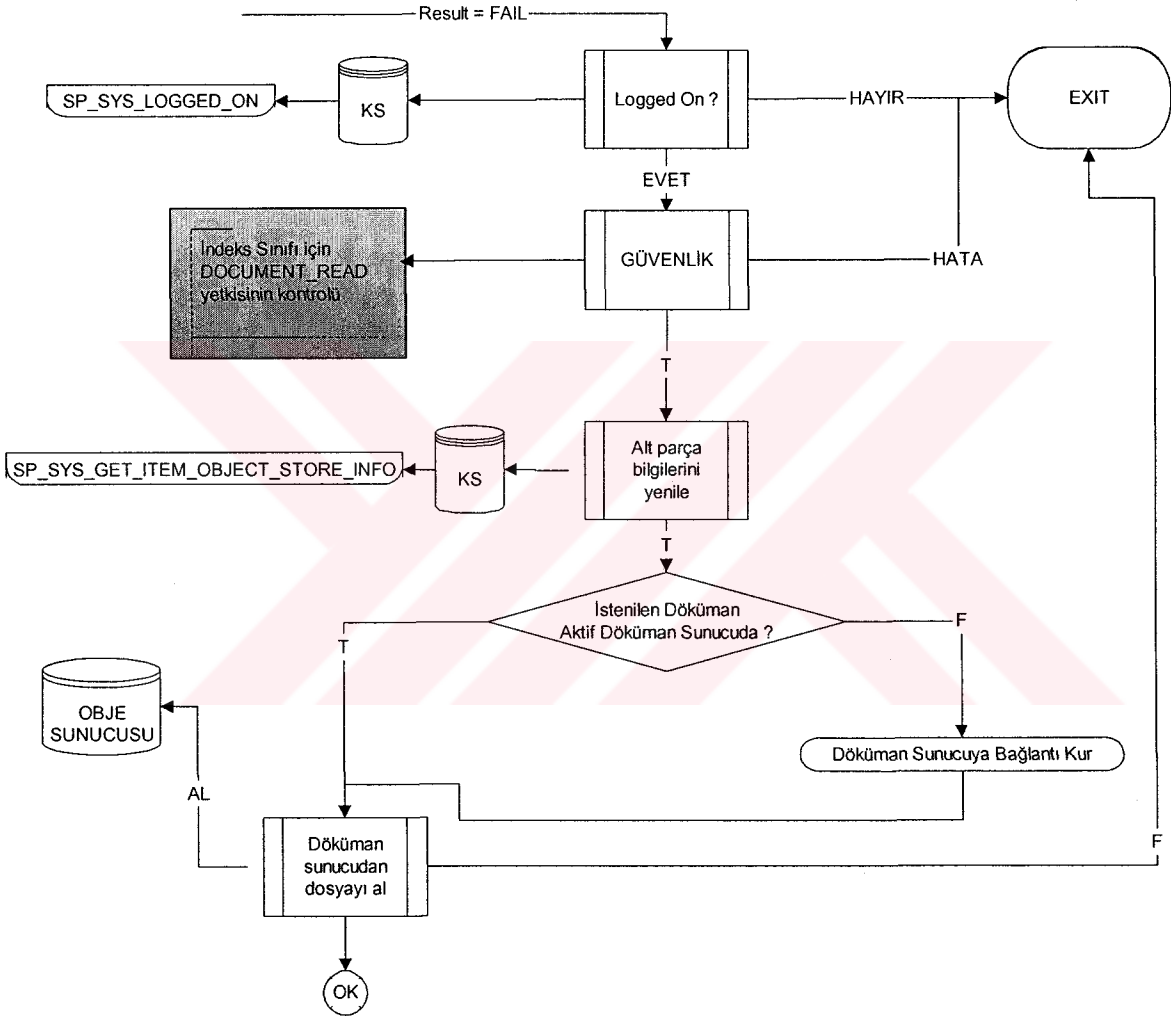


Şekil Ek.2-19 Getparentfolders metodu akış diagramı

GetPart (Döküman nesnesinin istenilen alt parçası ve versiyonunu döküman sunucudan getirir)

Giriş Parametreleri:
 PartNo(Integer) : Alt parça numarası
 VersionNo(Integer) : Versiyon no

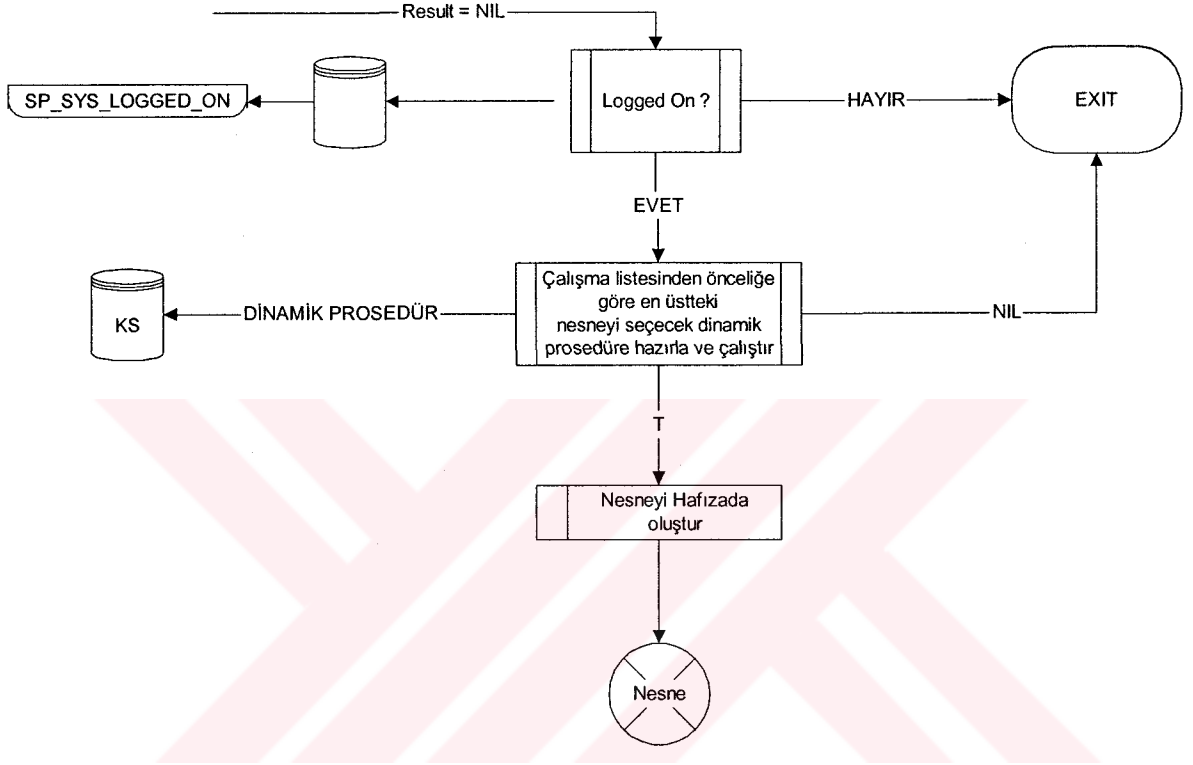
Çıkış Parametresi:
 ReturnVal (Integer) : 0 : OK, -1: FAIL



Şekil Ek.2-20 Getpart metodu akış diagramı

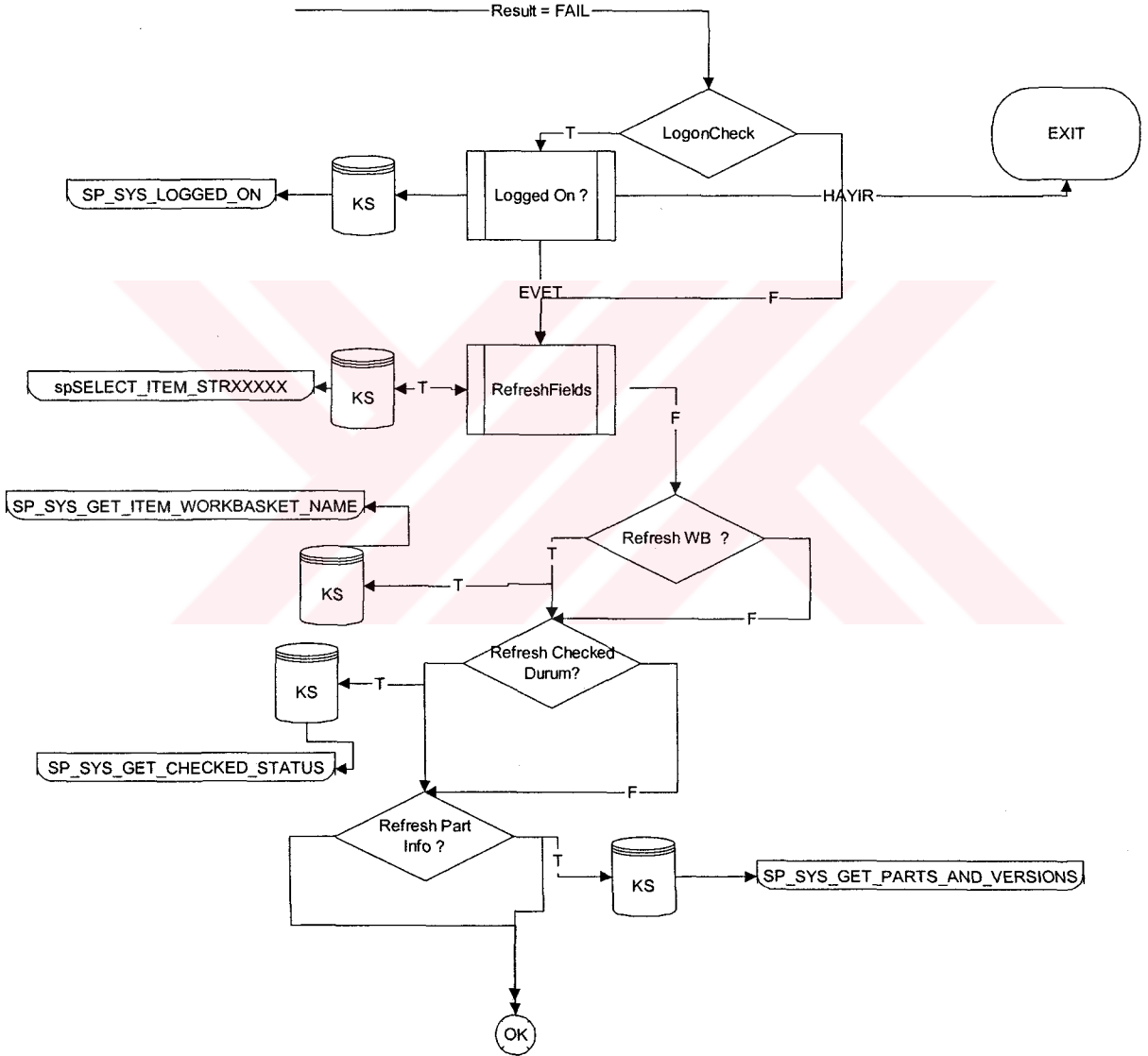
NextWorkBasketItem (Hızlı listelerde en öncelikli doküman veya klasörü döndürür.)

Çıkış Parametresi:
ReturnVal (İtem) : Nesne Objesi

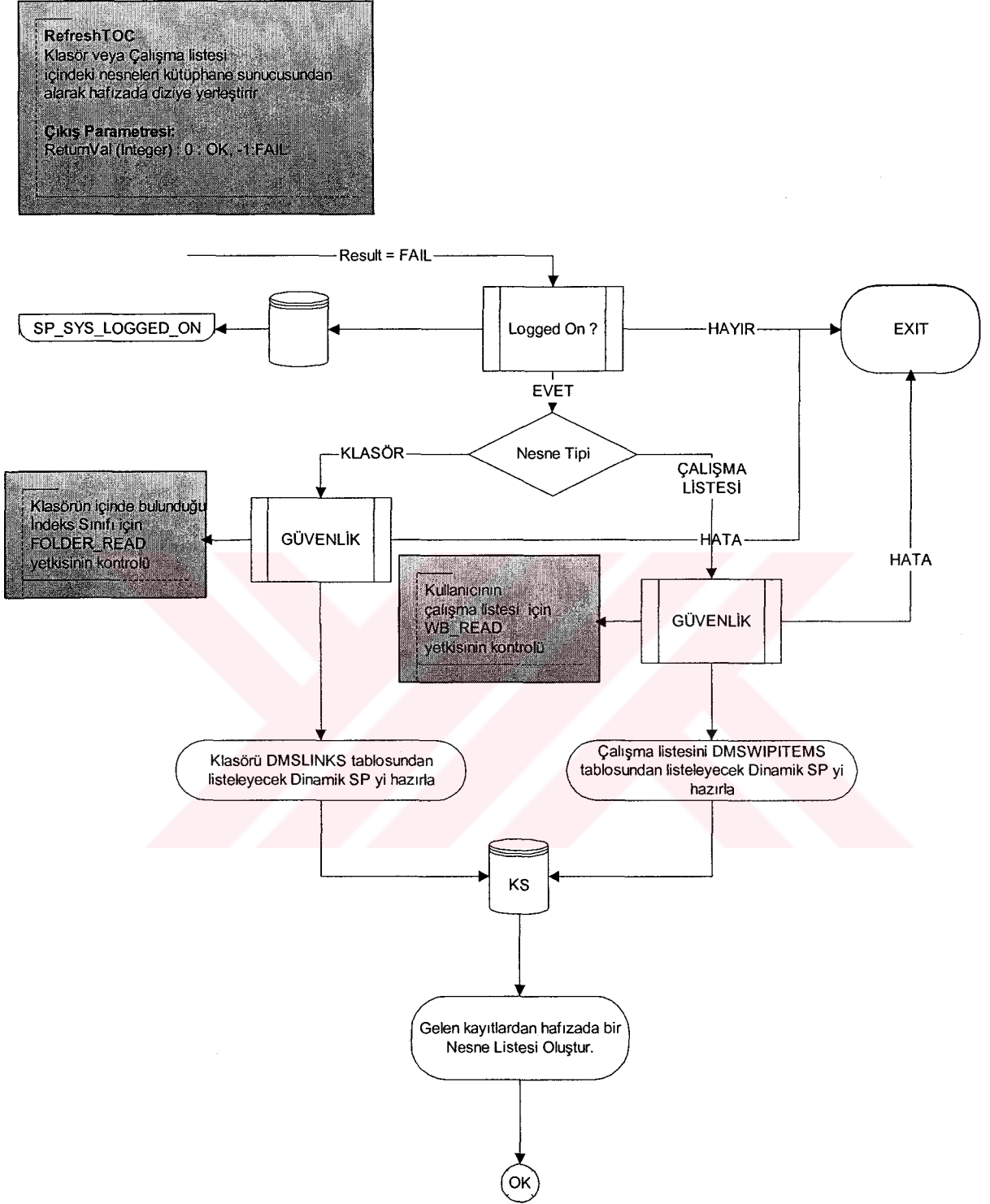


Şekil Ek.2-21 Nextworkbasketitem metodu akış diagramı

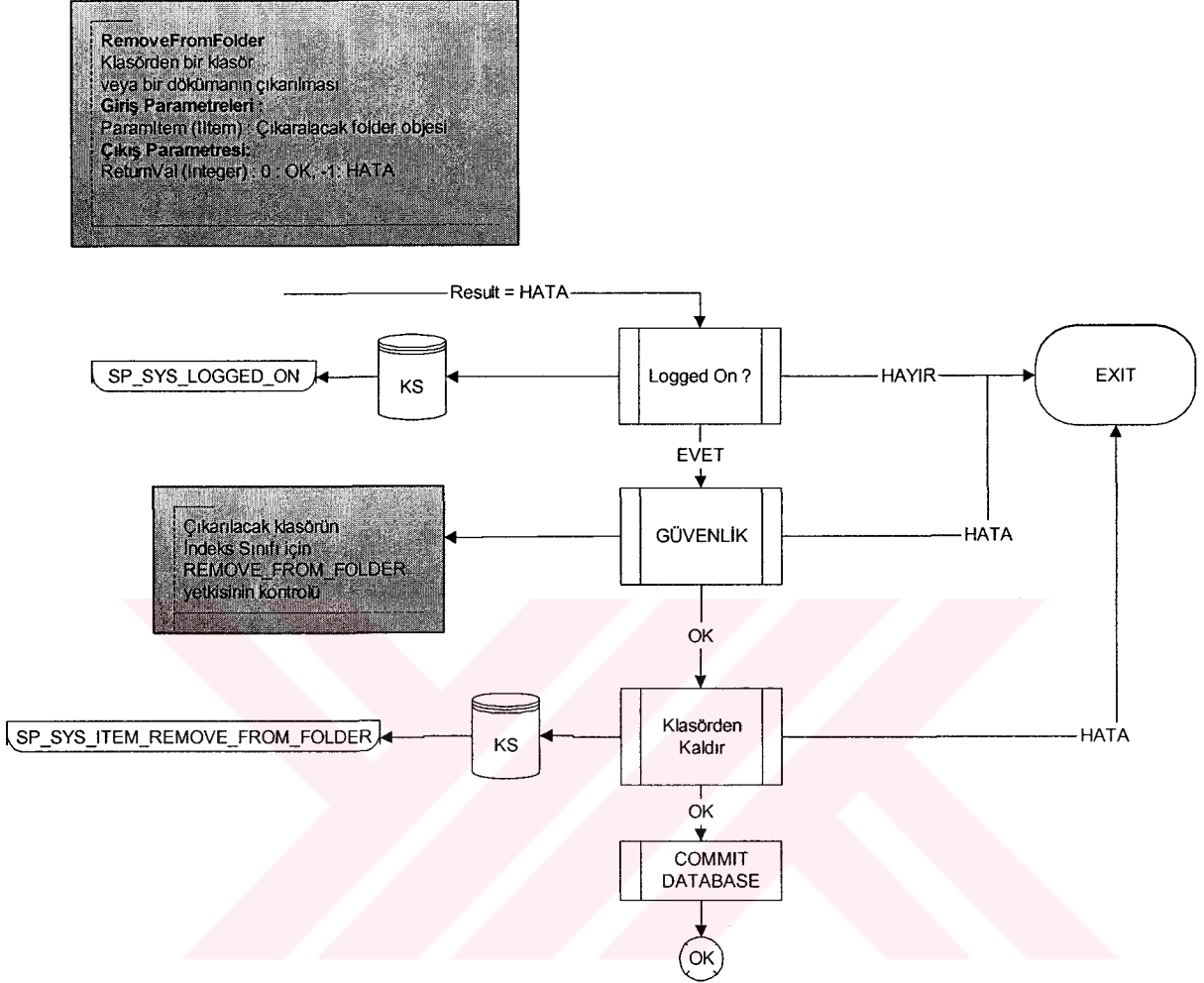
Open
 Klasör veya Dökümanı
 açmak ve bazı bilgileri tazelemek
Giriş Parametreleri :
 RefreshFields (bool) : İndeks bilgilerini tazele
 RefreshWB (bool) : Çalışma listesinin adının tazeleneceği
 RefreshPartInfo (bool) : Döküman alt parça bilgilerinin
 tazeleneceği
 RefreshCheckedStatus : Döküman veya klasörün check-out
 bilgisine bakılacak
 LogonCheck (bool) : Kullanıcının içinde logon olup
 olmadığına bakılacak
Çıkış Parametreleri :
 ReturnVal (Integer) : 0 - OK, 1-FAIL



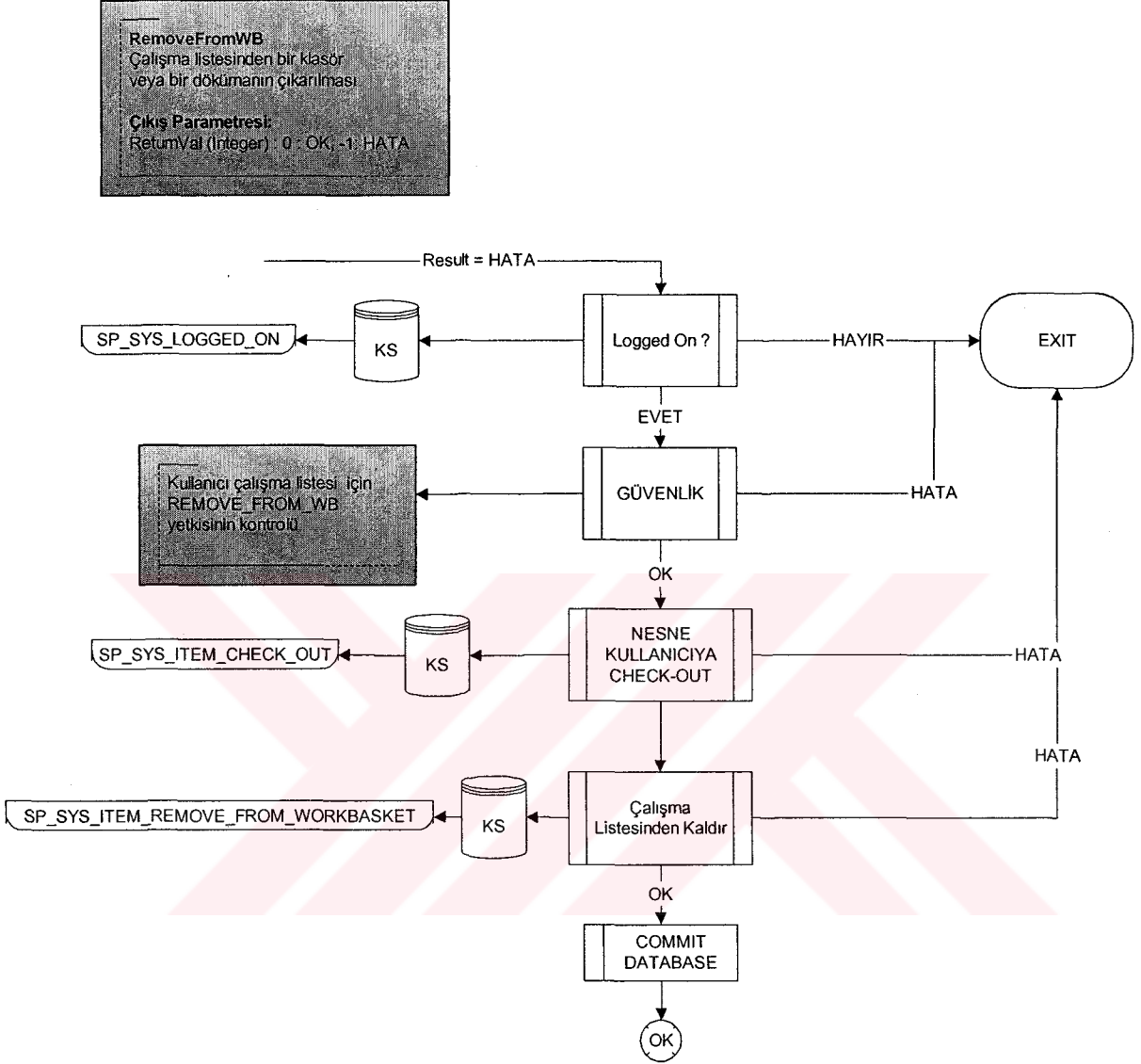
Şekil Ek2-.22 Open metodu akış diagramı



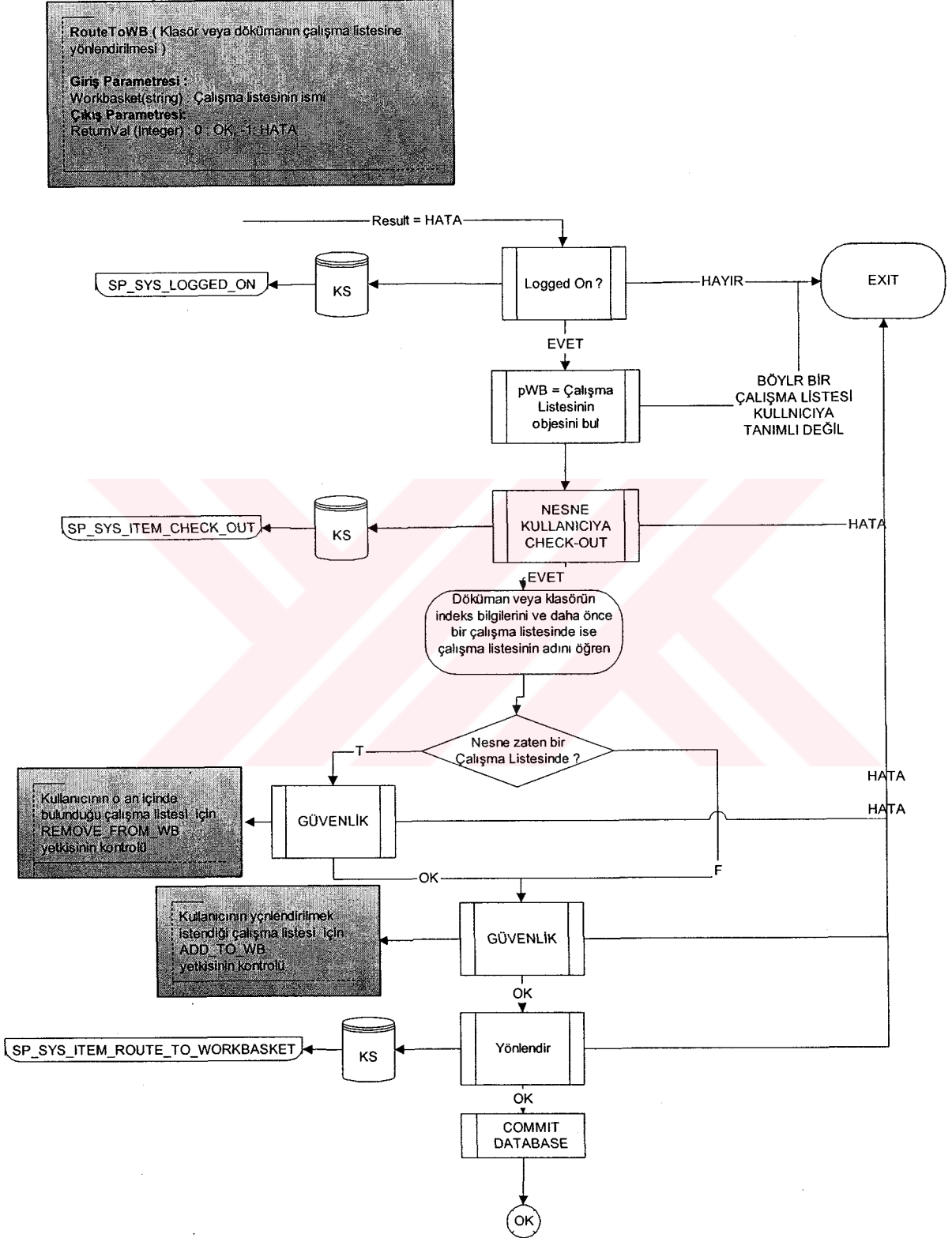
Şekil Ek.2-23 RefreshTOC metodu akış diagramı



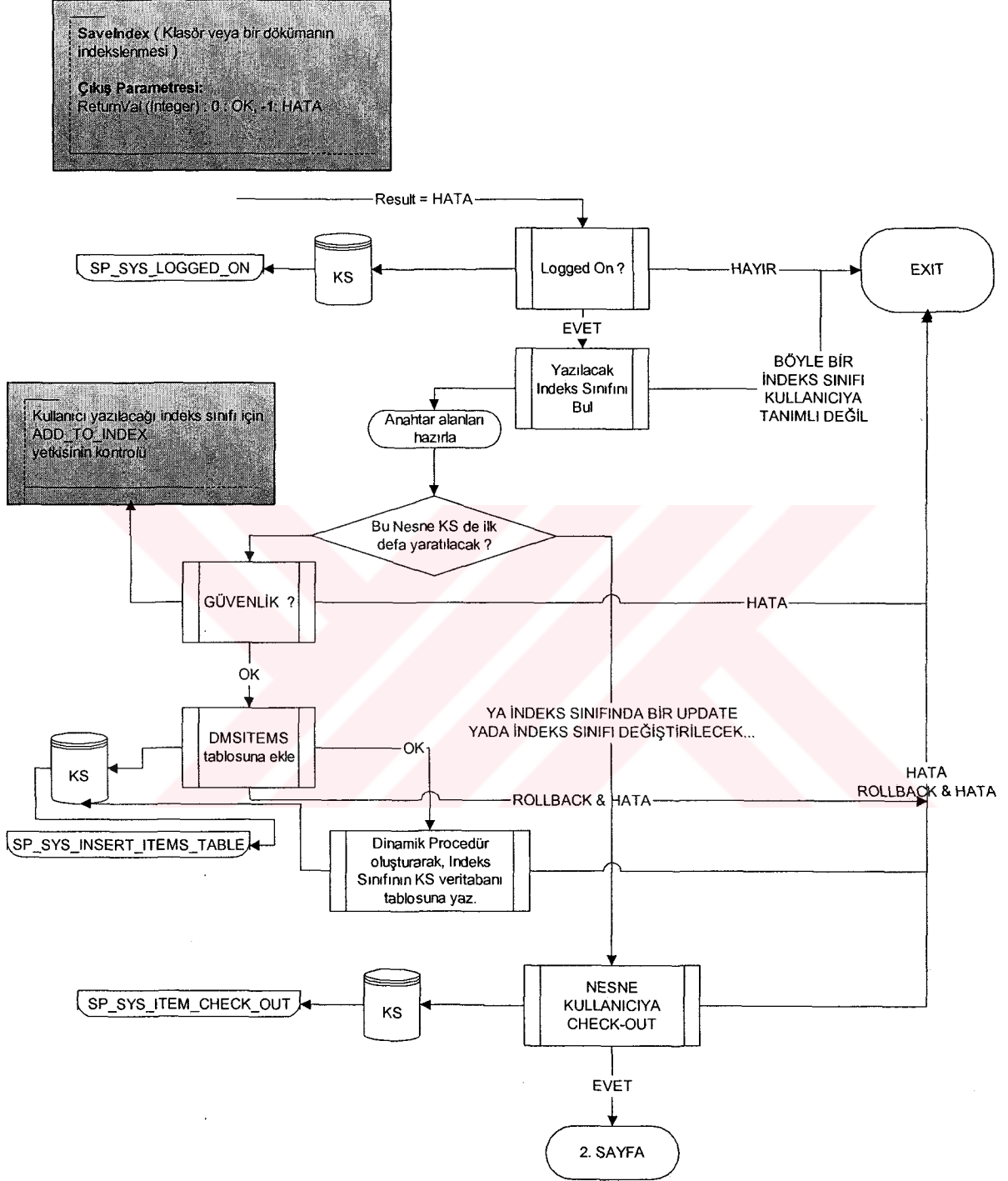
Şekil Ek.2-24 Removefromfolder metodu akış diagramı



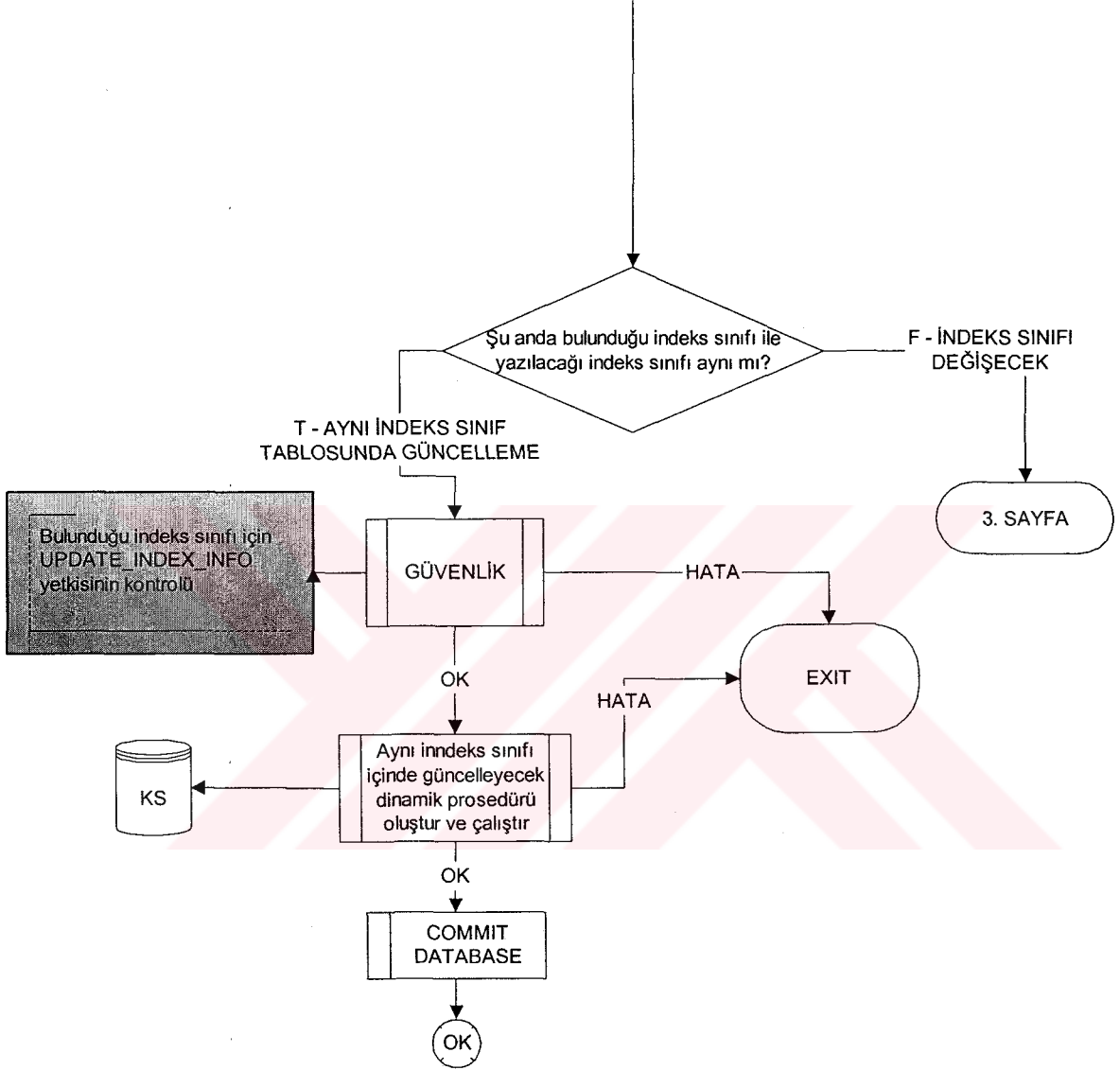
Şekil Ek.2-25 Removefromwb metodu akış diagramı



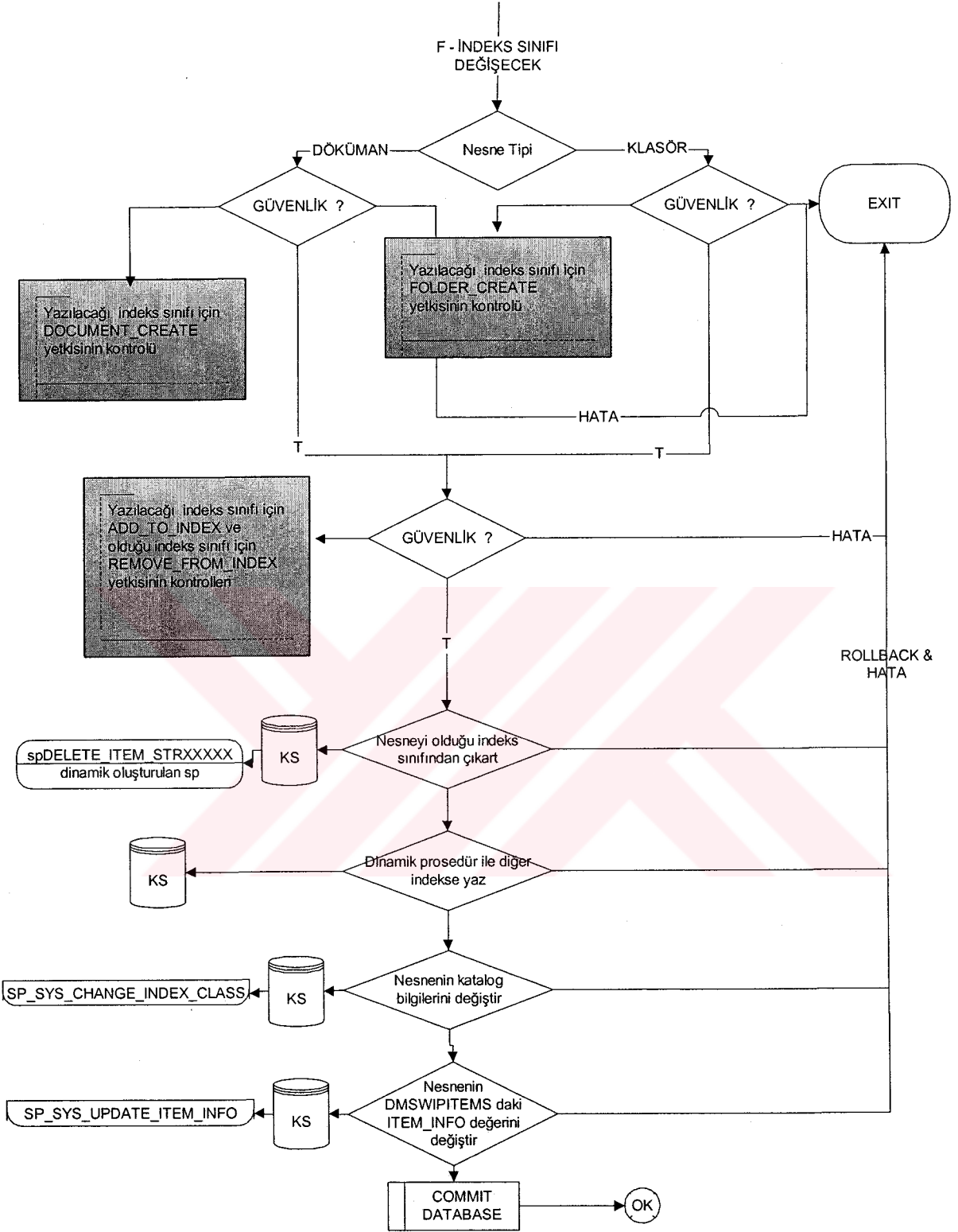
Şekil Ek.2-26 Routetowb metodu akış diagramı



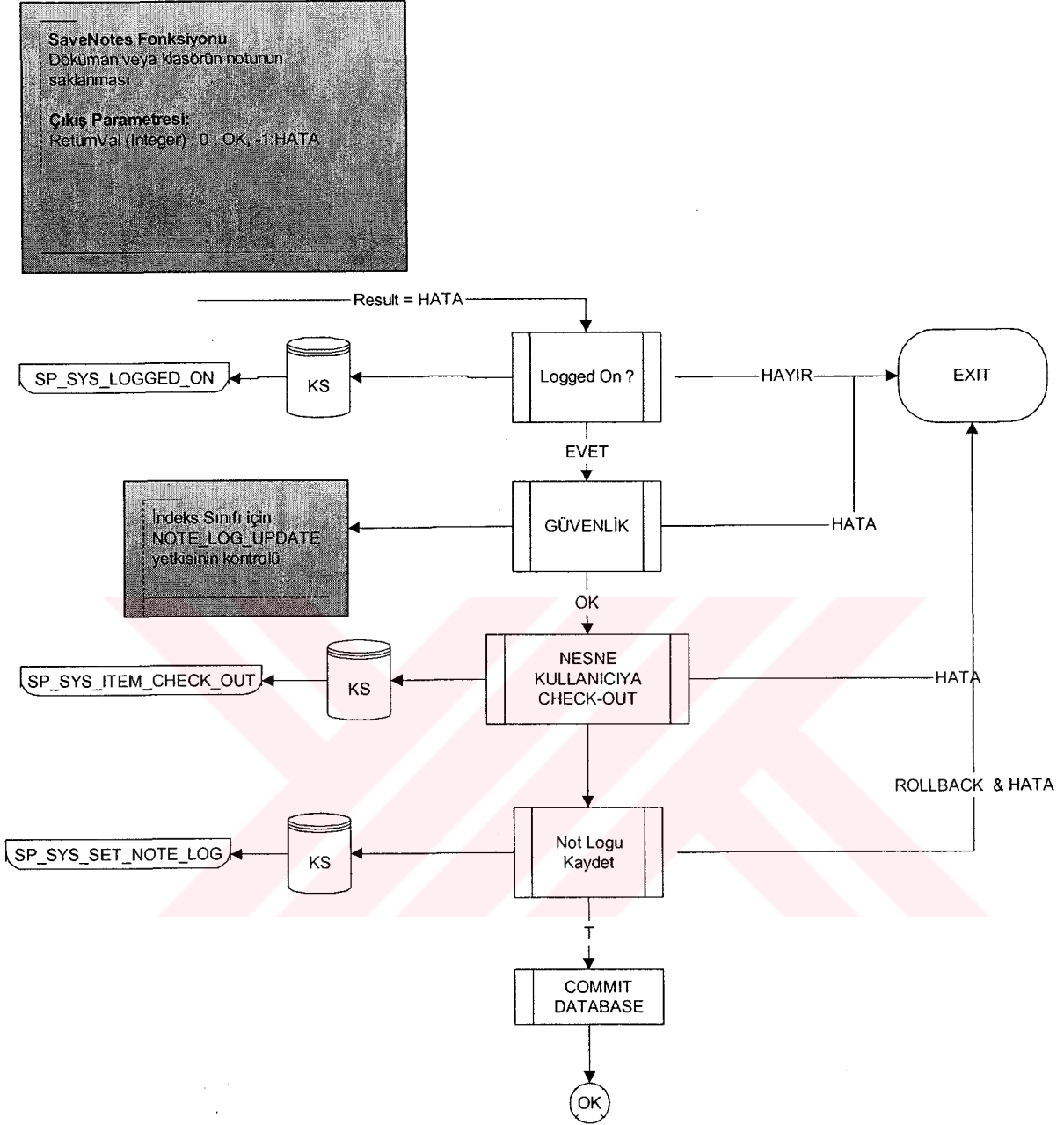
Şekil Ek.2-27 Saveindex metodu akış diagramı – 1



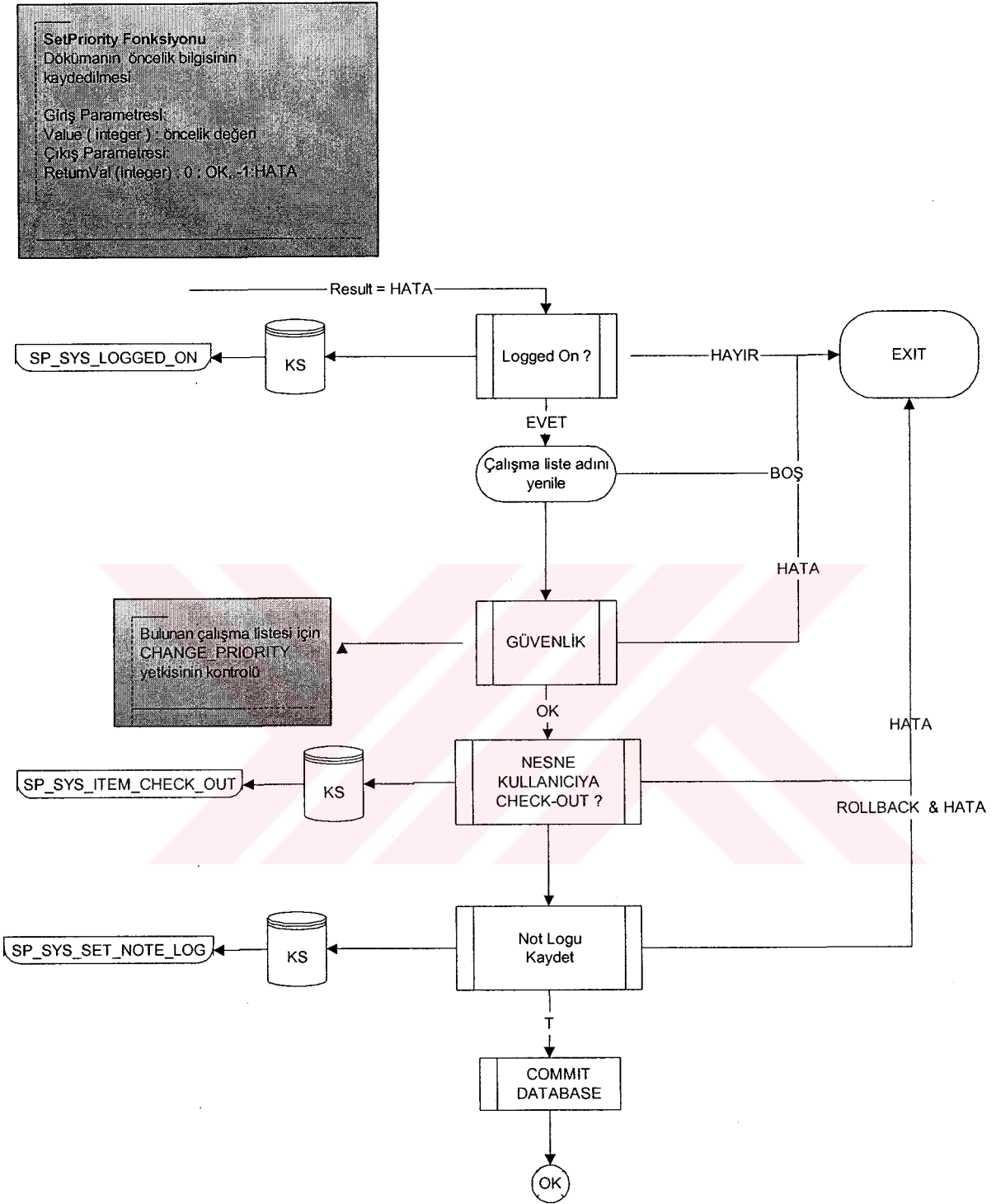
Şekil Ek.2-28 Saveindex metodu akış diagramı - 2



Şekil Ek.2-29 Saveindex metodu akış diagramı – 3



Şekil Ek.2-30 Savenotes metodu akış diagramı



Şekil Ek.2-31 Setpriority metodu akış diagramı

ÖZGEÇMİŞ

Doğum tarihi	17.03.1977	
Doğum yeri	İstanbul	
Lise	1991-1995	Kabataş Erkek Lisesi
Lisans	1995-1999	Yıldız Üniversitesi Elektrik-Elektronik Fak. Bilgisayar Bilimleri ve Mühendisliği Bölümü
Yüksek Lisans	1989-1991	Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Müh. Anabilim Dalı

Çalıştığı kurum(lar)

1999-Devam EdiyorGaranti Teknoloji

