

**YILDIZ TEKNİK ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ**

**ELEKTRONİK TİCARET**

Bilgisayar Mühendisi Perihan KİLİMCİ

F.B.E. Bilgisayar Bilimleri Mühendisliği Anabilim Dalında  
Hazırlanan

**YÜKSEK LİSANS TEZİ**

**106271**

Tez Danışmanı: Prof. M.Yahya KARSLIGİL

Prof. M. Yahya Karşılığ *Yahya*  
Doc. Dr. Oya Kalıpsız *Oya*  
Doc. Dr. Selim Akyokus *Selim*

İSTANBUL, 2001

**T.C. YÜKSEKÖĞRETİM KURULU  
DOKÜMANTASYON MERKEZİ**

106271

## İÇİNDEKİLER

KISALTIMA LİSTESİ.....	vi
ŞEKİL LİSTESİ.....	vii
ÇİZELGE LİSTESİ.....	ix
ÖNSÖZ .....	x
ÖZET .....	xi
ABSTRACT .....	xii
1 GİRİŞ .....	1
2 ELEKTRONİK TİCARET.....	4
2.1 Elektronik Ticaretin Kapsamı .....	5
2.2 Elektronik Ticaretin Tarafları.....	6
2.3 Elektronik Ticaretin Yararları .....	6
2.4 Elektronik Ticaretin Kategorileri .....	7
2.4.1 Tüketiciye yönelik elektronik ticaret.....	8
2.4.2 Kurumlar arası elektronik ticaret.....	9
2.4.2.1 Tedarik zinciri .....	10
2.5 Elektronik Ticaret Yönetimi Bileşenleri .....	12
2.6 Kurumların İnternette Sunabilecekleri Bilgiler.....	13
2.7 Dünyada Elektronik Ticaret .....	14
2.7.1 G7 ülkelerinde elektronik ticaret konusunda yapılan çalışmalar .....	14
2.7.2 Japonya’da elektronik ticaret konusunda yapılan çalışmalar.....	16
2.7.3 Avrupa Birliği’nde elektronik ticaret konusunda yapılan çalışmalar.....	16
2.7.4 Avustralya’da elektronik ticaret konusunda yapılan çalışmalar .....	17
2.8 Elektronik Ticaretin Güvenliği.....	17
2.8.1 Ülkelerin şifreleme politikalarındaki farklılıklar .....	18
2.9 Türkiye’deki Durum.....	19
3 İLGİLİ ÇALIŞMALAR.....	22
3.1 Kendi Arasında İşlenebilirlik .....	22
3.1.1 XML söz dizimi .....	24
3.1.1.1 Elemanlar .....	25
3.1.1.2 Başlangıç etiketleri .....	25
3.1.1.3 Bitiş etiketleri .....	26
3.1.1.4 Boş eleman etiketleri.....	26
3.1.1.5 Belge elemanı .....	26
3.1.1.6 Özellikler .....	27
3.1.1.7 Karakter veri.....	27
3.1.1.8 Karakter ve varlık referansları.....	28
3.1.1.9 Açıklamalar .....	28
3.1.1.10 İşleme komutları.....	28
3.1.1.11 CDATA bölümleri.....	29

3.1.1.12	İyi oluşturulmuş ve doğru belgeler.....	29
3.1.1.13	Belge tipi bildirimini .....	29
3.1.1.13.1	Elemanlar .....	31
3.1.1.13.1.1	Boş elemanlar.....	31
3.1.1.13.1.2	Yalnızca eleman içeren elemanlar.....	32
3.1.1.13.1.3	Karışık elemanlar .....	33
3.1.1.13.1.4	ANY elemanları .....	34
3.1.1.13.2	Özellikler.....	34
3.1.1.13.3	Varlıklar .....	37
3.1.1.13.3.1	Genel varlıklar.....	38
3.1.1.13.3.2	Parametre varlıklar .....	38
3.1.1.13.3.3	Karakter varlıklar .....	39
3.1.1.13.4	Gösterim .....	39
3.1.2	Document Object Model (DOM).....	40
3.2	Elektronik Kataloglar, Veritabanları ve Çoklu-Ortam Sistemleri.....	40
3.3	Akıllı Ajanlar.....	42
3.4	İş Akışları ve Süreç Otomasyonu.....	43
4	ELEKTRONİK TİCARET YAPILARI VE SİSTEMLERİ.....	45
4.1	Elektronik Ticaret Yapıları.....	45
4.1.1	eCo .....	45
4.1.1.1	Tasarım ilkeleri .....	47
4.1.1.2	Gerekenler .....	47
4.1.1.3	eCo yapısı.....	48
4.1.1.4	eCo yapı katmanları .....	49
4.1.1.4.1	CBL .....	52
4.1.1.5	eCo katmanları ve kayıtları .....	53
4.1.2	RosettaNet .....	55
4.1.3	BizTalk .....	58
4.1.3.1	BizTalk yapısı .....	58
4.1.3.1.1	XML .....	59
4.1.3.1.2	SOAP.....	59
4.1.3.1.3	BizTalk mesajları .....	60
4.1.3.1.3.1	Terminoloji.....	60
4.1.3.1.4	Mantıksal katmanlar.....	61
4.1.3.2	BizTalk.org.....	62
4.1.3.2.1	BizTalk şema tutucu.....	63
4.2	Elektronik Ticaret Sistemleri .....	64
4.2.1	Ajan aracılı elektronik ticaret.....	64
4.2.1.1	Kasbah.....	64
4.2.1.2	Tete-a-tete.....	65
4.2.2	Minnesota ajan pazar yeri mimarisi (MAGMA).....	65
4.2.3	Firefly .....	65
4.3	Antlaşma.....	66
4.3.1	Antlaşma sanat ve bilimi .....	66
4.3.2	Kendi kendini yürüten ajanlar için antlaşma kararı .....	67
4.3.3	Çoklu özelliklere göre antlaşmada yapay ajan öğrenme kuralları .....	67
4.3.4	Çok ajanlı sistemlerde artırma ile satış .....	67
5	WEB SERVİSLERİ .....	68
5.1	Web Ortamındaki Gelişmeler.....	69

5.2	Web Servisleri Modeli .....	71
5.3	Web Servisleri Standartları .....	73
5.3.1	SOAP.....	74
5.3.2	WSDL.....	76
5.3.3	UDDI.....	78
6	MICROSOFT BIZTALK SERVER 2000.....	80
6.1	BizTalk Mesajlaşma Servisleri.....	81
6.1.1	Organizasyonlar .....	81
6.1.2	Dokümanlar .....	82
6.1.3	Zarflar.....	82
6.1.4	Portlar ve Dağıtım Listeleri.....	82
6.1.5	Kanallar .....	82
6.2	Mesaj Transferi .....	83
6.2.1	Açık internet protokolleri .....	83
6.2.2	Windows'a özel haberleşme.....	83
6.3	BizTalk Server Araçları .....	84
6.3.1	Editör.....	85
6.3.1.1	WebDAV.....	85
6.3.2	Mapper .....	85
6.3.3	Orchestration Designer.....	86
6.3.3.1	Gerçekleştirim Teknolojileri .....	87
6.3.4	Server Administration .....	88
7	TASARIM.....	89
7.1	HKK.lığının Satıcı Kurumlardan Mal Alım Senaryosu .....	90
7.2	Alım Emri İsteği XML Dosyası.....	92
7.3	Alım Emri XML Dosyası.....	94
7.4	Fatura XML Dosyası.....	95
7.5	HKK Veritabanı .....	97
7.6	Satıcı1 Veritabanı .....	98
7.7	Kurumlar Arasındaki Mesajlaşma.....	99
7.7.1	Dokümanlar .....	99
7.7.2	Organizasyonlar .....	100
7.7.3	Portlar .....	100
7.7.3.1	T_Port AlEmIstToAlEm .....	101
7.7.3.2	T_Port HKKTKlfIste .....	102
7.7.3.3	T_Port S1HKKTKlfVer .....	102
7.7.3.4	T_Port S1TKlf.....	102
7.7.3.5	T_Port S1Teklif.....	102
7.7.3.6	T_Port S1AlEmToFat .....	103
7.7.3.7	T_Port S1SprsBilgEkle .....	103
7.7.3.8	T_Port S1HKKSprsVer.....	103
7.7.3.9	T_Port S1SprsV.....	103
7.7.3.10	T_Port S1Fat .....	104
7.7.4	COM parçaları.....	104
7.7.4.1	Basla.Tezim.....	104
7.7.4.2	HKKSpsS1.SprsS1 .....	104
7.7.4.3	HKKSpsS2.SprsS2.....	104
7.7.4.4	HKKSpsS3.SprsS3 .....	105
7.7.4.5	HKKTeklif.TeklifKaydet .....	105

7.7.4.6	HKK Tklif iste. Sprs .....	105
7.7.4.7	S1 Sprs HKKAIC. Satıcı 1 Sprs.....	105
7.7.4.8	S1 Sprs HKKQ. Sprs Schedule .....	106
7.7.4.9	S1 Teklif HKKQ. Tklif Schedule.....	106
7.7.4.10	S1 Teklif YazAIC. S1 Teklif.....	106
7.8	Parçaların BizTalk Orchestration Designer Aracı ile Bütünleştirilmesi .....	106
7.8.1	HKK lığının mal alım iş akışı .....	106
7.8.2	Satıcı 1'in teklif verme iş akışı.....	109
7.8.3	Satıcı 1'in sipariş alma iş akışı.....	110
8	UYGULAMA.....	112
8.1	Kullanılan Yazılımlar .....	112
8.2	Alıcı Kurum Tarafındaki Arayüzü ve İş Süreci .....	112
8.2.1	Çalışan sayfası ile yapılan işlemler .....	113
8.2.2	Mal sayfası ile yapılan işlemler.....	114
8.2.3	Kurum sayfası ile yapılan işlemler.....	115
8.2.4	Alım emri işlemi.....	116
8.2.5	Satıcı kurumların verdiği teklifler .....	117
8.3	Mal Alımında Alıcı-Satıcı Tarafında Çalışan Uygulamalar .....	118
8.4	Mesajlaşmada Kullanılan XML Dokümanları .....	122
8.5	İkinci Alıcı Kurum Tarafındaki Arayüzü ve İş Süreci.....	124
8.6	Birinci Satıcı Kurum Tarafındaki Arayüzü ve İş Süreci.....	125
8.6.1	Satıcı 1 kurumundan teklif istendiğinde çalışan uygulamalar .....	127
8.6.2	Satıcı 1 kurumuna sipariş verildiğinde çalışan uygulamalar.....	129
8.7	İkinci Satıcı Kurum Tarafındaki Arayüzü ve İş Süreci.....	130
8.8	Üçüncü Satıcı Kurum Tarafındaki Arayüzü ve İş Süreci.....	130
9	SONUÇLAR VE ÖNERİLER .....	132
KAYNAKLAR.....		134
EKLER .....		137
Ek 1	HKK Veritabanında Kullanılan Tablolar.....	138
Ek 2	Satıcı 1 Veritabanında Kullanılan Tablolar.....	139
Ek 3	DTD Tanımları.....	140
Ek 4	Hava Kuvvetleri Komutanlığı için ASP Kodları .....	147
Ek 5	Hava Kuvvetleri Komutanlığı için COM Kodları.....	197
Ek 6	Satıcı 1 için ASP Kodları .....	213
Ek 7	Satıcı 1 için COM Kodları .....	227
Ek 8	Windows Script Components.....	248
Ek 9	Organizasyon Tanımları.....	251
Ek 10	Dosya Alım Fonksiyonu Tanımları.....	252
Ek 11	Port Tanımları .....	253
Ek 12	Kuyruk Tanımları.....	255
Ek 13	ODBC Tanımları.....	256
Ek 14	Virtual Directory Tanımları .....	257
Ek 15	Elektronik Ticaret Terimler Sözlüğü (Bozkurt, 2000).....	258
ÖZGEÇMİŞ .....		262

## KISALTMA LİSTESİ

AIC	Application Integration Component
ANSI	American National Standard Institute
B2B	Business to Business
B2C	Business to Consumer
CBL	Common Business Library
C2C	Consumer to Consumer
COM	Component Object Model
DOM	Document Object Model
DTD	Document Type Definition
EDI	Electronic Data Interchange
EFT	Electronic Funds Transfer
GML	Generalized Markup Language
HTTP	Hypertext Markup Language
IBR	Intelligent Business Registries
IETF	Internet Engineering Task Force
ISO	International Organization for Standardization
ITU	International Telecommunication Union
KOBİ	Küçük ve Orta Boy İşletmeler
MIME	Multipurpose Internet Mail Extensions
MIT	Massachusetts Institute of Technology
MSMQ	Microsoft Message Queue
NAICS	North American Industry Classification System
OECD	Organisation for Economic Cooperation and Development
PIP	Partner Interface Processes
SCM	Supply Chain Management
SGML	Standardized Generalized Markup Language
SOAP	Simple Object Access Protocol
UDDI	Universal Description Discovery and Integration
UN/CEFACT	United Nations body for Trade Facilitation and Electronic Business
UNCITRAL	United Nations Commission on International Trade Law
UNCTAD	United Nations Conference on Trade and Development
UNTPDC	United Nations Trade Point Development Center
URI	Uniform Resource Identifier
W3C	World Wide Web Consortium
WebDAV	Web Distributed Authoring and Versioning
WSDL	Web Services Description Language
WTO	World Trade Organization
XML	Extensible Markup Language

## ŞEKİL LİSTESİ

Şekil 1.1	Forrester Research B2C-B2B tahminleri.....	1
Şekil 2.1	Web tabanlı ticaret modeli (B2B, business-to-business; B2C, business-to-consumer; M, marketplace).....	7
Şekil 2.2	Elektronik ticaret yönetimi bileşenleri.....	13
Şekil 3.1	XML örneği .....	24
Şekil 3.2	Harici belge tipi bildirimi.....	29
Şekil 3.3	Adres belgesinin DTD'si .....	30
Şekil 3.4	Karışık elemanın DTD'de tanımı.....	33
Şekil 3.5	Karışık elemanın XML ile gösterimi .....	33
Şekil 3.6	IDREF'in DTD'de tanımı .....	36
Şekil 3.7	IDREF'in XML ile gösterimi.....	36
Şekil 3.8	IDREFS'in DTD'de tanımı.....	37
Şekil 3.9	IDREFS'in XML'de gösterimi .....	37
Şekil 3.10	NMTOKENS'in DTD tanımı ve XML gösterimi.....	37
Şekil 3.11	DOM nesne yapısı.....	40
Şekil 4.1	eCo yapısının katmanları .....	49
Şekil 4.2	IBR bileşenleri .....	52
Şekil 4.3	eCo doküman örneği .....	53
Şekil 4.4	eCo uyumlu dokümanın kök elemanı .....	54
Şekil 4.5	RosettaNet cluster ve segment adları.....	56
Şekil 4.6	RosettaNet elektronik iş uygulama yapısı.....	57
Şekil 4.7	Protokol yapısı .....	57
Şekil 4.8	Mantıksal katmanlar.....	62
Şekil 5.1	Web servisleri ile etkileşim sağlayan iş gezisi uygulaması (Kaynak: IBM) ...	70
Şekil 5.2	Web servis modeli (Kaynak: IBM).....	71
Şekil 5.3	Web servisi istemci ve sağlayıcısı arasındaki temel işlemler (Kaynak: Sun)..	72
Şekil 5.4	Web servisi mimarisi katmanları (Kaynak: IBM) .....	73
Şekil 5.5	SOAP istemci-sunucu arasındaki iletişim (Kaynak: www.techmetrix.com)...	74
Şekil 5.6	Bir SOAP mesajının yapısı (Kaynak: www.techmetrix.com).....	75
Şekil 5.7	Bir SOAP istemci istek (request) mesajı .....	75
Şekil 5.8	Bir SOAP yanıt (response) mesajı .....	76
Şekil 5.9	Bir WSDL belgesi.....	78
Şekil 5.10	UDDI kurum kayıt sunucuları .....	79
Şekil 6.1	BizTalk Server'in doküman işleme süreci.....	86
Şekil 7.1	Ağ şeması.....	90
Şekil 7.2	HKKlılığı-Satıcı Kurum arasındaki dosya akış şeması.....	91
Şekil 7.3	Kurumlar arası mesajlaşma.....	91
Şekil 7.4	Alım emri isteği XML dosyasında tutulan veriler.....	92
Şekil 7.5	Alım emri isteği dokümanının DTD'si .....	94
Şekil 7.6	Alım emri XML dosyasında tutulan veriler.....	94
Şekil 7.7	Fatura XML dosyasında tutulan veriler.....	96
Şekil 7.8	HKK veritabanı Varlık-İlişki şeması .....	97
Şekil 7.9	Satıcı1 veritabanı Varlık-İlişki şeması.....	98
Şekil 7.10	Organizasyonlar arasında mesajlaşma ve kullanılan portlar.....	101
Şekil 7.11	Alım Emri İsteğinin Alım Emrine çevrimi .....	107
Şekil 7.12	Alım Emrinin Faturaya çevrimi.....	108
Şekil 7.13	HKK.skv iş akışı .....	109
Şekil 7.14	Satıcı1 kurumunun, Hava Kuvvetleri Komutanlığı için teklif iş akışı.....	110

Şekil 7.15	Satıcı1 kurumunun, Hava Kuvvetleri Komutanlığı için sipariş iş akışı.....	111
Şekil 8.1	Hava Kuvvetleri Komutanlığı arayüzü .....	113
Şekil 8.2	Çalışan personel bilgileri sayfası .....	114
Şekil 8.3	Mal bilgileri sayfası .....	115
Şekil 8.4	Kurum bilgileri sayfası .....	116
Şekil 8.5	Alım emri sayfası.....	117
Şekil 8.6	Hava Kuvvetleri Komutanlığına verilen teklifler .....	118
Şekil 8.7a	HKKlığının teklif alımı sırasında çalışan uygulamalar .....	119
Şekil 8.7b	HKKlığının teklif alımı sırasında çalışan uygulamalar .....	119
Şekil 8.8	HKKlığının sipariş verip, fatura alımına kadar çalışan uygulamalar .....	121
Şekil 8.9	Alım emri isteği XML dokümanı .....	122
Şekil 8.10	Alım emri XML dokümanı .....	122
Şekil 8.11	Sipariş aşamasında fatura XML dokümanı.....	123
Şekil 8.12	Fatura aşamasında fatura XML dokümanı.....	124
Şekil 8.13	Kara Kuvvetleri Komutanlığı arayüzü.....	125
Şekil 8.14	Satıcı1 teklif sayfası.....	126
Şekil 8.15	Satıcı1 sipariş sayfası.....	127
Şekil 8.16	Satıcı1 kurumundan teklif istendiğinde çalışan uygulamalar .....	128
Şekil 8.17	Satıcı1 kurumuna sipariş verildiğinde çalışan uygulamalar .....	129
Şekil 8.18	Satıcı2 web sayfası.....	130
Şekil 8.19	Satıcı3 web sayfası.....	131



## ÇİZELGE LİSTESİ

Çizelge 2.1	Elektronik Ticaret: İnternet temelli satışlarda büyüme.....	8
Çizelge 3.1	DTD işaretleme ifadeleri.....	31
Çizelge 3.2	DTD sembol anlamları.....	33
Çizelge 3.3	DTD default değer tipleri.....	34
Çizelge 3.4	DTD özellik tipleri.....	35



## ÖNSÖZ

Tez çalışmam sırasında sürekli olarak beni yönlendiren ve araştırmaya sevk eden tez danışmanım Prof. M.Yahya KARSLIGİL'e teşekkür ederim.

Kurumlar arası elektronik ticaret uygulaması için kullanmam gereken yazılım konusunda beni yönlendiren ve tez çalışmam sırasında olumlu düşünceleriyle beni destekleyen Prof.Dr.Asuman DOĞAÇ'a, tezim için gerekli yazılımı sağlayan Microsoft'tan Tülin OKTAY Hanım'a teşekkür ederim.

Kurumlar arası elektronik ticaret uygulaması geliştirmemde fikirleriyle tezimi yönlendiren, kavramları daha net ortaya koymama yardım eden, Bilişim 2001'de kısa bir sunuş yapmamı sağlayan, tezimi hazırlamam sırasında beni yönlendiren ve teşvik eden Doç.Dr.Selim AKYOKUŞ'a teşekkür ederim.

İlk akademik makale yazmamda katkıları olan komutanım, Hv.Y.Müh.Alb. Dr.Sefer KURNAZ'a teşekkür ederim.

Her zaman beni destekleyen anneme, babama, kardeşim Tolga KİLİMCİ'ye, Hv.Y.Müh.Ütğm.Suna Mutlu DÜZ'e, Hava Harp Okulu Bilgi İşlem Müdürlüğüne ve Hava Kuvvetleri Komutanlığına teşekkür ederim.

## ÖZET

Elektronik ticaret, internet üzerindeki tüketiciye ve kurumlara yönelik tüm işlemleri kapsamaktadır. Dünya ve Türkiye’de tüketiciye yönelik elektronik ticaret çok yaygın olarak gerçekleştirilmektedir. Kurumlar arası bilgi değişimi ve bilgiyi işleme standart olmadığı için kurumlar arası elektronik ticaret çok az sayıda gerçekleştirilmiştir. Bu nedenle günümüzde, kurumlar arasındaki teknolojik uyumu sağlamak, globalleşen dünyamızda önemli bir problem olarak karşımıza çıkmaktadır.

Kurumlar arası uyumu sağlayabilmek için çok çeşitli elektronik ticaret yapıları geliştirilmektedir. Bu kurumlar arası elektronik ticaret yapıları arasında eCom, RosettaNet ve BizTalk’un özellikle incelenmesi gerekmektedir. Bu yapılar temel olarak XML (eXtensible Markup Language)’i almaktadır. XML, Java’nın sunduğu taşınabilir kod gibi, bilgi dünyasına taşınabilir veriyi sunarak büyük bir ilerleme elde edilmesini sağlamıştır. Böylelikle kod, veri ve veriyi sunuş teknikleri farklı sınıflara ayrılmıştır.

Bu tez çalışması ile eCom, RosettaNet ve BizTalk kurumlar arası elektronik ticaret yapıları incelenmiş ve Hava Kuvvetleri Komutanlığı, Kara Kuvvetleri Komutanlığı ile 3 satıcı kurum arasında kurumlar arası elektronik ticaret modeli tasarlanmış, tasarlanan kurumlar arası elektronik ticaret modelini gerçekleştirmenin kolay olmadığını göstermek amacıyla bu alım-satım işlemi, gelişmekte olan Microsoft BizTalk Server 2000 ile modellenmiştir.

Kurumlar arası elektronik ticaret yapılarını tek bir standartta toplayabilmek için RosettaNet’in sunduğu kurumlar ve ürünler sözlüğü tüm endüstriler için gerçekleştirmeli, eCom yapısı gibi marketin dinamik olması sağlanmalı ve BizTalk’un sunduğu yazılım parçaları içerilmelidir. Ancak bu şekilde bütünleşik bir kurumlar arası elektronik ticaret sistemi kolayca sunulabilir.

**Anahtar Kelimeler:** Kurumlar arası elektronik ticaret, elektronik ticaret, iş-akışları, ajanlar, RosettaNet, eCom, BizTalk, XML.

## ABSTRACT

Electronic Commerce includes all online transactions from businesses to the consumers and businesses on the internet. Business-to-Consumer electronic commerce is implemented tremendously however, cause of data transfer and data processing technics are not standard between businesses, Business-to-Business electronic commerce is implemented in small numbers in Turkey and World. Thus, today in our global world interoperability between businesses is the most important problem.

Many frameworks are developed to solve the problem of interoperability between businesses. eCom, RosettaNet and BizTalk frameworks are the most promising ones. All of these frameworks are based on the documents, XML (eXtensible Markup Language). Like Java which is serving portable code on the internet, XML is serving portable data on the internet. With these emerging technologies code, data and data presentation technics are classified into different categories.

eCom, RosettaNet, BizTalk frameworks are searched and Business-to-Business electronic commerce is designed to model buyer-seller processes between Turkish Air Force Commandant, Turkish Army Force Commandant and three seller firms. After that, designed model is implemented in Microsoft BizTalk Server 2000 to demonstrate how difficult to perform Business-to-Business electronic commerce.

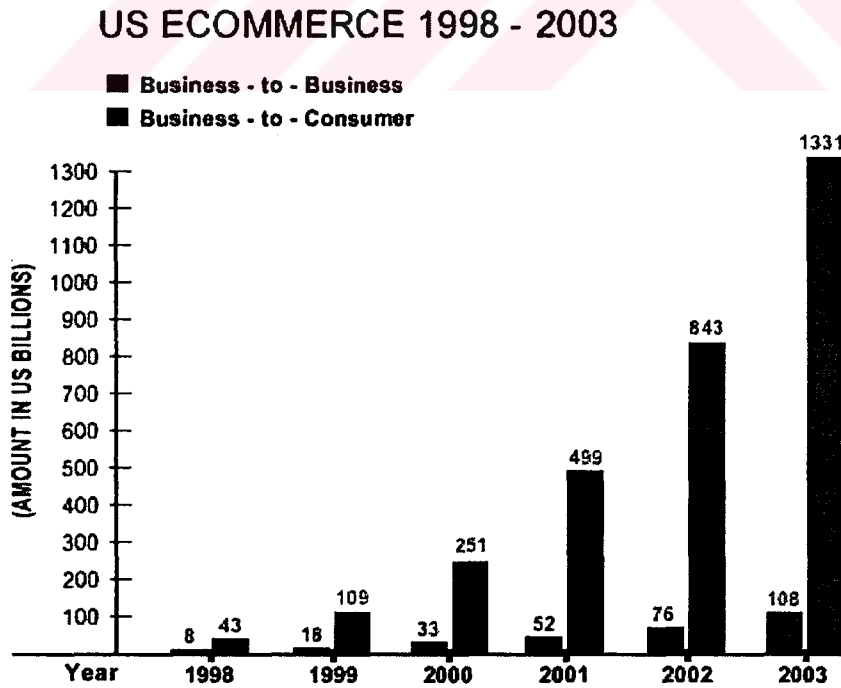
As a result, an integrated Business-to-Business electronic commerce, should include all markets like RosettaNet 's solution, should be dynamic like eCom framework and should have implementation details like BizTalk framework. Only with this way market can be implemented easily.

**Keywords:** Business-to-Business, electronic commerce, workflow management, agents, RosettaNet, eCom, BizTalk, XML.

## 1 GİRİŞ

İnternet, dijital TV, mobil sistemler gibi herhangi bir bilgi ya da iletişim teknolojisi aracılığı ile gerçekleştirilen satış işlemleri elektronik ticareti oluşturmaktadır (7). Elektronik ticaret; kurumdan kuruma, kurumdan tüketiciye, kurumdan yönetime (şirket ve hükümet kurumlarına) ve tüketiciden yönetime gerçekleştirilmektedir. Günümüzde kurumdan tüketiciye (B2C) elektronik ticaret, Türkiye’de ve dünyada yaygın olarak gerçekleştirilmektedir. B2C elektronik ticaret; yeni satış kanalları yaratması, mevcut pazar potansiyelini genişletmesi ve yenilerini yaratması açısından önem taşımaktadır.

Yüzlerce ortakla iş yapan kurumlar ve endüstriler, internetin sağladığı avantajları kullanarak daha fazla kar elde etmek istemektedir. Bu nedenle kurumlar arasında yapılan elektronik ticaret günümüzde daha çok önem kazanmıştır. Forrester Research araştırma şirketine göre, belirli bir aşamadan sonra B2B elektronik ticaret hızlı ve çok büyük bir büyüme kazanacaktır. Yaptıkları araştırmaya göre, ABD’de B2B elektronik ticaret 1998 yılından itibaren beş yılda her yıl iki kat büyüyecek ve 2003 yılında 1.3 trilyon dolarlık bir hacme sahip olacaktır (Akyokuş, 2001). Şekil 1.1, B2B-B2C tahminlerini göstermektedir.



Şekil 1.1 Forrester Research B2C-B2B tahminleri

Ortak iş ve ticaret yapan kurumlar, farklı endüstriler farklı bilgi sistemlerini kullanmaktadır. Bu durumda B2B elektronik ticaretin gerçekleştirilmesi için kurumlar, aralarında geçen

mesajlaşmayı ve veri işleme tekniklerini standartlaştırmak zorundadır. Kurumlar arası uyumluluğu sağlayabilmek için uluslararası kurumlar, bir çok standart geliştirmiştir. Bu standartlar arasında XML-tabanlı B2B yapıları; eCom, RosettaNet ve BizTalk önemlidir.

RosettaNet bilgi teknolojisi (Information Technology), elektronik parçalar (Electronic Components) ve yarıiletken üretimi (Semiconductor Manufacturing) tedarik zincirleri için standartlar geliştirmektedir. Fakat bu şekilde standartlaşma, arayüzlerin sınırlı olması nedeniyle, dinamik bir elektronik ticaret sağlamamaktadır. Ayrıca sadece teknoloji sektörüne yönelik çalışmalar yaptığı için, kurumlar arası ticareti tam olarak gerçekleştirememektedir. eCom ise katmanlı bir yapı ile kurumları modellemiştir. Bu katmanlarda yapılan işler, tip kayıtlarında (type registries) tutulmaktadır. Bu sayede kurumlar arası ticaretin dinamikliği sağlanmaktadır. BizTalk'un sunduğu yapı ise standartları, işakışlarını, ajan yapısını, yazılım detaylarını içeren bütünleşik bir sistemdir.

BizTalk yapısı, Microsoft BizTalk Server 2000 ile gerçekleştirilmektedir. Diğer B2B yapıları ise böyle bir olanak sunmamakta, sadece tasarım detaylarını vermektedir. Microsoft BizTalk Server 2000 ürünü, Aralık 2000'de satışa sunulmuştur. Bu ürün, kurumların aralarında geçen tüm mesajlaşmayı kaydetmeye olanak sağlamaktadır. Kurumlar arasındaki iletişimler, XML dokümanları ile şema saklayıcıda (schema repository) tutulmaktadır. İleride bu şemalar ile kurumlar arası ticaretin standartlaşması planlanmaktadır. Microsoft BizTalk Server 2000'in içindeki araçlar, kurumlar arası ticareti modellemek için yeterli olmaktadır. Özellikle BizTalk Mapper gibi bir doküman eşleyici, diğer kurumlar arası ticaret modeli geliştirmeyi sağlayan yazılımlar tarafından sunulmamaktadır. Tasarım detayı olarak, Microsoft Message Queue (MSMQ)'nun içerilmesi ürünü daha da çekici yapmıştır.

Basit bir kurumlar arası elektronik ticaret modeli oluşturmak, kurumlar arası uygulama geliştirmede ortaya çıkabilecek güçlükleri görmek, iş süreçlerini modellemek, mesajlaşma yapılarını tanımlamak için kamu kurumu ile satıcı kurumlar arasında geçen kurumlar arası elektronik ticaret sistemi Microsoft BizTalk 2000 Server kullanılarak geliştirilmiştir. Bu uygulamada, 2 alıcı kurum, Hava Kuvvetleri Komutanlığı ve Kara Kuvvetleri Komutanlığı; satıcı kurumlar olarak; Satıcı1, Satıcı2 ve Satıcı3 kurumları seçilmiştir. Bu sistemde, alıcı kurumun yapabileceği işlemler; alım emri isteğini oluşturmak, alım emri isteğini alım emrine çevirmek, oluşturulan alım emrini satıcı kurumlara göndermek; satıcı kurumlardan gelen teklifleri kaydetmek, bu teklifler arasında en ekonomik olanı seçen alım kararına göre uygun

satıcı kuruma sipariş vermek, siparişi kazanan kuruma göndermektir. Satıcı kurumların yapabileceği işlemler ise; teklif oluşturmak, oluşturulan teklifleri alıcı kurumlara göndermek, sipariş almak, fatura göndermektir.

Bu doğrultuda, tez çalışmasının dokuz bölüm altında toplanması uygun görülmüştür.

İkinci bölümde, elektronik ticaretin; tanımı, kapsamı, tarafları, yararları, kategorileri, yönetim bileşenleri, dünya üzerindeki ve Türkiye üzerindeki durumu incelenmiştir.

Üçüncü bölümde, elektronik ticaretin dayandığı teknolojiler tanımlanmış, XML (eXtensible Markup Language), DTD (Document Type Definition), DOM (Document Object Model), elektronik kataloglar, veritabanları ve çoklu ortam sistemleri, akıllı ajanlar, iş akışları ve süreç otomasyonu açıklanmıştır.

Dördüncü bölümde, eCom yapısı, RosettaNet yapısı, BizTalk yapısı, dünyadaki elektronik ticaret sistemleri, alım kararı verme metotları açıklanmıştır.

Beşinci bölümde, yeni bir kavram olması nedeniyle; SOAP (Simple Object Access Protocol), WSDL (Web Services Definition Language), UDDI (Universal Description Discovery and Integration) teknolojileri üzerine kurulmuş olan web servisleri açıklanmıştır.

Altıncı bölümde, Microsoft BizTalk Server 2000 tanıtılmıştır. Kurumlar arası elektronik ticareti geliştirmemizi sağlayan araçları açıklanmıştır.

Yedinci bölümde, Hava Kuvvetleri Komutanlığı, Kara Kuvvetleri Komutanlığı ve 3 satıcı kurum arasında geçen kurumlar arası elektronik ticaret tasarlanmıştır.

Sekizinci bölümde, yedinci bölümde tasarlanan kurumlar arası elektronik ticaret modeli, kurumlar arası ticareti gerçekleştirmenin ne kadar zor olduğunu göstermek amacıyla Microsoft BizTalk Server 2000 ile gerçekleştirilmiştir.

Dokuzuncu bölümde, elde edilen sonuçlar sunulmuştur.

## 2 ELEKTRONİK TİCARET

Elektronik ticaret, basit olarak “elektronik aygıtlar aracılığı ile yapılan ticaret” şeklinde tanımlanmaktadır. Bu tanımı biraz daha genişletirsek: “İnternet, dijital TV, mobil sistemler gibi herhangi bir bilgi ya da iletişim teknolojisi aracılığı ile gerçekleştirilen satış işlemleri elektronik ticareti oluşturur” (7).

Bozkurt’a (2000) göre elektronik ticaret süreci, müşterilerin ürün bilgilerine erişmelerine olanak sağlama, satın alınacak malları seçme, malları güvenli olarak satın alma ve satın almayı finansal olarak sonuçlandırma aşamalarından oluşmaktadır.

Elektronik ticaretin bir çok kurum ve kişiler tarafından tanımı yapılmıştır. Bu yüzden elektronik ticaretin ne olduğu konusunda tam bir görüş birliğine ulaşılmış değildir. Kimileri, elektronik araçlarla yapılan, kurumlar arası elektronik veri değişimi (EDI), elektronik fon aktarımı (EFT) ve kredi-borç kartlarının kullanımı gibi tüm ticari işlemleri elektronik ticaret saymaktadır. Kimileri de, internet gibi açık ağlar üzerinde işlem ve ödemeleri yapılan, tüketicilere yönelik perakende ticareti elektronik ticaret olarak kabul etmektedir (Bozkurt, 2000).

Elektronik ticaret konusunda, dünya genelinde yaşanan gelişmelere paralel olarak bir çok ülke ve uluslararası kurum elektronik ticaret ve onunla bağlantılı konuları gündemine almıştır. **WTO** (Dünya Ticaret Örgütü), **UNCTAD** (BM Ticaret ve Kalkınma Konferansı), **OECD** (İktisadi İşbirliği ve Kalkınma Teşkilatı), **Avrupa Birliği**, Birleşmiş Milletlere bağlı **UN/CEFACT** (Birleşmiş Milletler Yönetim, Ticaret ve Ulaştırma İşlemlerini Kolaylaştırma Merkezi), **UNCITRAL** (Birleşmiş Milletler Uluslararası Ticaret Kanunu Komisyonu) gibi kurumlar ile **ITU** (Uluslararası Haberleşme Birliği) ve **Dünya Bankası**’nın son bir kaç yıldır ticarete etkinliğin sağlanması amacıyla, elektronik ticarete ilişkin konularda yoğun olarak çalışmaya başladıkları gözlenmektedir (7).

Elektronik ticaret, uluslararası organizasyonlar ve bu alanda faaliyet gösteren bazı uluslararası kurumlar tarafından aşağıdaki şekilde tanımlanmıştır.



## Elektronik ticaret;

- **WTO:** Mal ve hizmetlerin üretim, reklam, satış ve dağıtımlarının telekomünikasyon ağları üzerinden yapılmasıdır.
- **OECD:** Sayısallaştırılmış yazılı metin, ses ve görüntünün işlenmesi ve iletilmesine dayanan kişileri ve kurumları ilgilendiren tüm ticari işlemlerdir.
- **UN/CEFACT:** İş, yönetim ve tüketim faaliyetlerinin yürütülmesi için yapılanmış ve yapılanmamış iş bilgilerinin, üreticiler, tüketiciler ve kamu kurumları ile diğer organizasyonlar arasında elektronik araçlar (Elektronik posta ve mesajlar, elektronik bülten panoları, www teknolojisi, akıllı kartlar, elektronik fon transferi, elektronik veri değişimi vb.) üzerinden paylaşılmasıdır.

### 2.1 Elektronik Ticaretin Kapsamı

Yukarıdaki tanımlardan elektronik ticaretin kapsamını, şu şekilde belirleyebiliriz: elektronik ortamda **açık ve kapalı ağlar üzerinden** yapılan;

Mal (taşınır, taşınmaz) ve hizmet (bilgi servisleri, danışmanlık, finans, hukuk, sağlık, eğitim, ulaştırma vb.) ticareti, sayısal biçime çevrilmiş yazılı metin, ses, video görüntülerinin işlenmesi ve iletilmesi, ürün tasarımı, üretim, doğrudan tüketiciye pazarlama, üretim izleme, sevkiyat izleme, tanıtım, reklam ve bilgilendirme, sipariş verme, sözleşme yapma, banka işlemleri ve fon transferi, gümrükleme, mühendislik ve ortak tasarım geliştirme, kamu alımları, elektronik (sanal) para çıkarma, elektronik hisse alışverişi ve borsa, açık arttırma, sayısal imza, e-noterlik, güvenilir üçüncü taraf işlemleri, vergilendirme ve vergi toplama, fikri mülkiyet haklarının transferi, kiralanması vb. işlemleri elektronik ticaret kapsamında değerlendirilmektedir (7).

## 2.2 Elektronik Ticaretin Tarafları

Elektronik Ticaretin taraflarını şöyle sıralayabiliriz (7):

- Alıcı,
- Satıcı,
- Üretici,
- Bankalar,
- Komisyoncular,
- Sigorta şirketleri,
- Nakliye şirketleri,
- Özel sektör bilgi teknolojileri,
- Sivil toplum örgütleri,
- Üniversiteler,
- Onay kurumları, elektronik noterler,
- Dış Ticaret Müsteşarlığı,
- Gümrük Müsteşarlığı,
- Diğer kamu kurumları.

## 2.3 Elektronik Ticaretin Yararları

Elektronik ticaret, işlem maliyetlerini düşürmenin yanında, piyasanın rekabete uygun ve etkin çalışmasını kolaylaştırma gibi daha bir çok yararlar getirmektedir. Bunlardan bazıları şu şekilde sıralanabilir:

- Firmalarda örgütlenme ve yönetim etkinliğini artırır,
- Piyasayı firmalar bakımından genişletir. Bunun sonucunda, alıcılar ve satıcılar açısından seçme olanaklarını artırır,
- Tüketicilere, ürünler hakkındaki bilgileri tam ve mükemmel aktarır,
- Üretim ve dağıtım zincirini küçültür,
- Fiyat ve alım-satımın diğer koşullarını, karşılaştırma olanağı yaratır,
- Yeni ürün ve hizmetleri, tüketicilere ulaştırmak kolaylaşır,
- Piyasaya üreticilerin girişini kolaylaştırır ve rekabeti artırır,

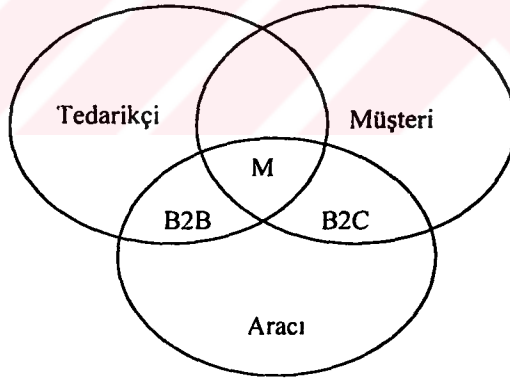
- Yeni faaliyetlerin gelişmesini ve verimliliğin artmasını sağlar. Böylece yeni nitelik gerektiren emeğe talep oluşur ve yeni iş alanları yaratılır,
- Dünya ekonomilerinin önemli bir parçası olan KOBİ (Küçük ve Orta Boy İşletmeler)'in küresel ticarete katılmaları kolaylaşır.

## 2.4 Elektronik Ticaretin Kategorileri

Elektronik ticaretin kategorileri dört başlık altında toplanmaktadır (Bozkurt, 2000). Bunlar;

- Kurumlar arası
- Kurumdan tüketiciye,
- Kurumdan yönetime (şirket ve hükümet kurumlarına),
- Tüketiciden yönetime elektronik ticaret şeklindedir.

Graham ve Hardaker'e (2000) göre web tabanlı ticaret ayrımı Şekil 2.1'de verilmiştir.



Şekil 2.1 Web tabanlı ticaret modeli (B2B, Business-to-Business; B2C, Business-to-Consumer; M, Marketplace)

Bu kategorilerden özellikle kurumlar arası elektronik ticaret, bugün olduğu gibi, gelecekte de, elektronik ticaretin en önemli boyutunu oluşturmaktadır.

Dünyada ticaret hacmi yılda % 69 büyümektedir. OECD ve IDC gibi kuruluşlara göre, 1997 yılında 11 milyar olarak gerçekleşen elektronik ticaret hacminin, 2003 yılında 400 milyarı

aşması ve 2005 yılında da 1 trilyon doları bulması beklenilmektedir. Bu durum Çizelge 2.1’de gösterilmiştir (Bozkurt, 2000).

Çizelge 2.1 Elektronik Ticaret: İnternet temelli satışlarda büyüme

Yıllar	Kurumlar arası(milyar \$)	Kurumdan Tüketickiye(milyar \$)
1997	7	5
1998	22	12
1999	47	22
2000	97	37
2001	176	59
2003	338	94

Hangi temel yaklaşımla olursa olsun, elektronik ticaret yapısı bir bütündür ve her yöntem de birbirleriyle iç içe yer almaktadır. Herhangi bir amaca yönelik olarak gerçekleştirilen elektronik ticaret sisteminin stratejik çalışması, diğerinden ayrı olarak ele alınamaz ya da değerlendirilemez.

#### 2.4.1 Tüketicilere yönelik elektronik ticaret

Son kullanıcılara yönelik elektronik ticaret çalışmaları, hedefin bireysel tüketiciler olduğu Business to Consumer (B2C) olarak isimlendirilen çalışmalardan oluşmaktadır. Kurumlar arasında gerçekleştirilen elektronik ticaret hacmine göre daha düşük bir kapasitesi bulunmaktadır. Bunun başlıca nedenleri, PC kullanımının tüketiciler arasında çalışma ortamına oranla daha kısıtlı olması, internet üzerindeki güvenlik sistemlerine kuşkulu yaklaşım ve tüketicilerin elektronik ticaret ile elde edebilecekleri kazanç hakkında yetersiz bilgilendirilmesidir.

Elektronik ticaret, yalnızca yeni bir satış kanalı oluşturmaktan çok, gerek mevcut pazar potansiyelini genişletmesi ve gerekse de yenilerini yaratması açısından önem taşımaktadır. İnternetin getirdiği iletişim avantajı, satıcılarla alıcıların tek bir platformda ve kendi belirledikleri ölçüde iletişime geçebilmelerine olanak tanımıştır. Özellikle son kullanıcıları

hedefleyen kurumların örnek aldıkları en önemli çalışmalardan olan Amazon ([www.amazon.com](http://www.amazon.com)), Cdnnow ([www.cdnnow.com](http://www.cdnnow.com)), Tesco ([www.tesco.com](http://www.tesco.com)), OnSale ([www.onsale.com](http://www.onsale.com)), Expedia ([www.expedia.com](http://www.expedia.com)) sözü edilen potansiyeli gözler önüne sermiştir (Dolanbay, 2000).

1998 yılında, internet kullanıcıları (B2C), 8 milyar dolarlık alışveriş gerçekleştirmişlerdir ve bu rakamın 2003 yılına kadar 140 milyar dolara ulaşması beklenmektedir.

Kurumdan-tüketiciye elektronik ticarete büyüme, genellikle eğlence, seyahat, habercilik, finans hizmetleri ve e-posta gibi nesnel olmayan ürünlerde görülmektedir.

Kurumların tüketicilere yönelik olarak gerçekleştirdikleri çalışmalarda beklenen asıl gelişme, dijital TV ve mobil sistemlerin kullanımının yaygınlaşması ile yaşanacaktır.

#### **2.4.2 Kurumlar arası elektronik ticaret**

Kurumlar arası elektronik ticaret (B2B), ticari kurumlar arasında geleneksel yollarla gerçekleştirilen iş ilişkilerinin elektronik ticaret yolu ile yapılması olarak tanımlanabilir. Kurumlar arası elektronik ticaret, günümüzde gerçekleştirilen toplam internet ticaretinin %80'ini oluşturmakta ve bu çalışmaları gerçekleştiren başarılı kurumlar bulunmaktadır. Bunlar arasında ağ sistemlerine yönelik çözümler üreten Cisco Systems ([www.cisco.com](http://www.cisco.com)), elektronik ticaret yapısında yıllık 2 milyar dolar ciro sağlamaktadır.

Geleneksel kurumlar arası elektronik ticaret, son yıllarda üç yönde biçimlenmektedir. Birincisi, www sayfalarının ve öbür ağların kullanımına dayalı olarak kurum, internet üzerinde genişlemektedir; ikincisi, kurum kendi iç işleyişini ya da işlerini, bir iç ağ ile görmekte yani içsel ağ sistemi oluşturmaktadır; üçüncüsü de, iç ağı, iş yaptığı öbür kurumlara bağlamakta, dışsal ağ düzeni kurmaktadır. Dışsal ağ, zincirleme bir yapılanmayla, kurum ile iş yapan ya da birlikte çalışan ikinci, üçüncü kuşak kurumlara bağlanmaktadır.

Bu üç yönlü elektronik ticaretin her bir bölümü, kurumun örgütlenmesi, istihdam yapısı, ürün kalitesi ve buradan işlem maliyetleri üzerinde etkili olmaktadır. Ancak, bu etkileri, doğrudan ve dolaylı biçimleriyle sayıya dökmek hiç de kolay değildir. Bu nedenle elektronik ticaret ile

satışlar ve ekonomik büyüme üzerindeki etkileri konusunda en azından şimdilik sınırlı ölçüde sonuç alınabilmekte ya da yalnızca yorumlarla yetinilmektedir.

Kurumlar arası elektronik ticaretin yaygınlaşmasına neden olan en önemli etkenler, işlem masraflarının azalması ürün ya da hizmetin kalitesinin yükselmesi ve tüketici hizmetlerinin iyileşmesidir. Kurumlar arası ticaret sayesinde Tedarik Zincirleri de önem kazanmıştır.

#### 2.4.2.1 Tedarik zinciri

Tedarik Zinciri (Supply Chain), herhangi bir ürün ya da hizmetin geliştirilmesi, üretilmesi ya da sunulması şeklinde ortaya çıkarılması, satış kanallarına aktarılması ve en sonunda da tüketiciye ulaştırılması olarak tanımlanabilir.

Hem servis, hem de üretim alanında kritik önem taşıyan Tedarik Zinciri Yönetimi (Supply Chain Management-SCM), sektörler arasında önemli farklılık taşıdığı gibi, kurumlar arasında da uygulama ve prensipler açısından farklılıklar da bulunmaktadır.

Tedarik zincirlerinin en önemli bileşenlerinden birisi, benzer özellikler içeren üretim ve sunum kriterlerinin tek bir ortamda değerlendirilebilmesidir. Bu değerlendirme sürecinde yararlanılacak kaynaklar arasındaki farklılık, benzer ürünlerin farklı kaynaklardan sağlanabilmesi, değişik dağıtım kanalları ile ürün ve hizmetlerin devam edebilme özellikleri, Tedarik Zinciri Yönetiminin elektronik ticaret sistemlerinde de çok önemli bileşenlerden birisi olarak çıkmasını sağlamıştır.

Geleneksel olarak, pazarlama, dağıtım, üretim ve satın alma işlemlerinin organizasyonu, tedarik zincirinin temel halkalarını oluşturmaktadır. Her kurumun kendi açısından belirlediği öncelik ve hedefler doğrultusunda bu aşamaların tanımlarında uygulama farklılıkları görülebilir. Zincirde yer alan her bir halkanın diğerini doğrudan ilgilendirmesi, yönetimin tek bir kritere bağlı olarak gerçekleştirilememesine neden olmaktadır. Örneğin, pazarlama stratejilerinin belirlenmesi, ürünün özellikleri ile ilgili olacaktır ya da fiyatlandırma politikası, satın alma aşamasından etkilenecektir.

Tedarik Zinciri Yönetimi, kurumun organizasyonel yapısına doğrudan bağımlı bir çalışmadır. Kurum bünyesinde yer alan her çalışma sistemi, bu yönetimi doğrudan ilgilendirmektedir.

Özellikle küçük ve orta ölçekli kurumlar açısından tedarik zinciri, farklı kurumların bir arada değerlendirilmesi gereken bir konu olarak ortaya çıkmaktadır. Bu zincirde yer alan kurumlar birbirinden bağımsızdır ve lojistik, üretim, hammadde sağlama ya da satış gibi belirli isteklere yönelik hizmet sunmaktadırlar. Bu tür yapılarda Tedarik Zinciri Yönetimi, kurumlar arası çalışmaların da yönetimini içine almaktadır. Özellikle koordinasyon yetersizliği, bu tür çalışmalarda önemli aksaklıklara neden olmaktadır.

Tedarik Zinciri Yönetimi, stratejik ve operasyonel olmak üzere iki farklı disiplin olarak ele alınmalıdır. Stratejik yönetim, adından da çıkarılabileceği gibi, kurumun çalışma stratejileri ile doğrudan ilgilidir ve bunların tedarik zinciri ortamında yeniden yorumlanması olarak ortaya çıkmaktadır. Yönetimin bu bölümü, uzun süreli hedeflere yönelik tanımlanmıştır ve belirleyici kriterler kurum yapısı ile ilintilidir. Operasyonel yönetim, kısa dönemli hedeflere yöneliktir ve ürünün satış kanalları, müşteriye iletilmesi ya da hammaddenin satın alındığı diğer şirketler gibi, dönemsel çalışmalar ışığı altında sürekli değiştirilebilir.

Tedarik Zinciri Yönetimi, kurumların yıllardır sürdürdükleri çalışmalardandır. İnternet ile birlikte, diğer tüm çalışmalarda olduğu gibi, Tedarik Zinciri Yönetimi de farklı çalışmaları içermeye başlamış ve bu alanda da önemli fırsatlar ortaya çıkmıştır. İnternet ekonomisinin yeniden şekillendirdiği çalışmalar ve getirdiği fırsatlar sonucunda, kurumların; tüketiciler, birlikte çalıştığı şirketler, satıcılar ve personel ile iletişimi dünya geneline yayılmıştır. Bu çerçevede kurumlar, bağımsız çalışmalar yerine diğer kurumlar ile birlikte çalışmalara girişmişlerdir. Bütün kurumların ortak hedefi, internet ile birlikte ulaşılabilen kriterler içeriyor olması nedeniyle maliyete odaklanmaktır.

Bu kriterlerde Tedarik Zinciri Yönetiminin önemi şu başlıklar halinde ortaya çıkmaktadır (7):

- Coğrafi sınırlardan bağımsızlaşma sonucu üretim yerlerinin belirlenmesinde ortaya çıkan esneklik,
- Ulaştırma maliyetlerinin azalması,
- Tedarikçilerin daha geniş yelpazeden seçilebilmesi,

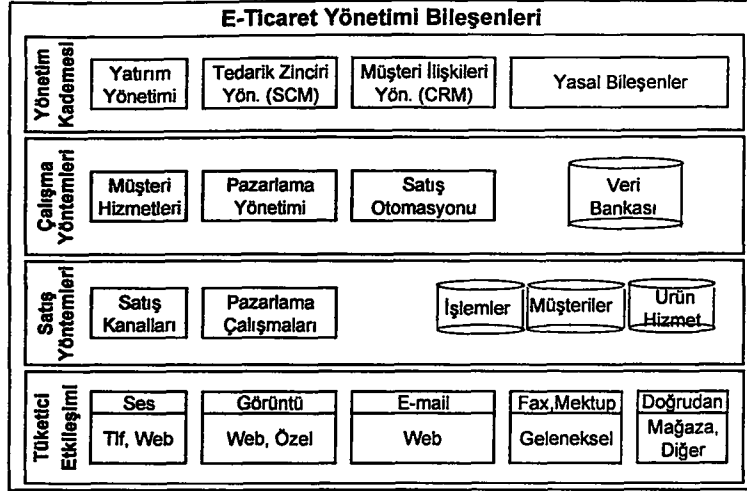
- Personelin farklı kanallar aracılığı ile sağlanması ve böylece maliyetinin düşürülebilmesi,
- Tüketicilere farklı kanallardan ulaşılarak kullanıcılara yönelik harcamalarda (destek ve tanıtım gibi) elde edilen önemli azalmalar,
- Vergilendirmelerin ülkelere göre farklılık göstermesi sonucunda, belirli çalışmaların farklı ülkelere kaydırılabilmesi.

Elektronik ticaret, özellikle kurumlar arası biçimiyle, maliyetleri düşürmektedir. Alt yapısına erişim olanağı ucuzladıkça ve kolaylıkla yaygınlaştıkça, elektronik ticaretin, mal ve hizmet üretimi, girdi-çıkıtı stoklarının ya da envanterinin tutulması süreci, satışlar, siparişlerin yerine getirilmesi ve son olarak da alıcıya/tüketicie sunulan hizmetlerde büyük maliyet düşüşleri sağlayacağına kesin gözüyle bakılmaktadır. Bununla birlikte maliyet düşüşleri; fiyatları belirleyen piyasa yapısı ya da rekabet koşulları, tüketicinin alım gücü ve eğilimleri olduğundan fiyatlara bire bir yansıtılamamakta, yine de elektronik ticaretin, fiyatlama yapısını değiştireceği düşünülmektedir.

## **2.5 Elektronik Ticaret Yönetimi Bileşenleri**

Pazaryerinin değişmesi, genel olarak kurumların yapısının, özellikle de satın alma, üretim, satış ve istihdam örgütlenmelerinin değiştirilmesine yol açmaktadır. Bu süreç, kurumların, yönetimleriyle birlikte, çalıştırdıkları işgücünün çok yönlü nitelikte olmasına, esnekliğe ve sürece uyum sağlamasına bağlıdır. Elektronik ticaretin kurumları başarıya götürmesi için yönetime ve kurumun çalışma yöntemlerine tam olarak uyumlu olması gerekmektedir (Bozkurt, 2000). Bu nedenle, elektronik ticaretin yönetim bileşenlerini belirlemek önem kazanmaktadır (Dolanbay, 2000). Şekil 2.2'de elektronik ticaret yönetimi bileşenleri verilmiştir.





Şekil 2.2 Elektronik ticaret yönetimi bileşenleri ( Dolanbay, 2000)

## 2.6 Kurumların İnternette Sunabilecekleri Bilgiler

İnternette sunulabilecek bilgiler, aşağıdaki başlıklar altında ele alınabilir (Dolanbay, 2000).

- Satış ve Pazarlamaya Yönelik olarak;
  - Ürün broşürleri,
  - Elektronik kataloglar,
  - Ürün özellikleri ve kullanım kılavuzları,
  - Promosyon ve özel indirim bilgileri,
  - Fiyatlandırma politikaları,
  - Sıklıkla karşılaşılan sorunlar ve çözümleri,
  - Online sipariş ve ödeme hizmetleri,
  - Etkileşimli iletişim,
  - Gerekli olabilecek bağlantılar,
  - Çoklu ortam tanıtımları.
- Kullanıcı desteğine yönelik olarak;
  - İletişim bilgileri,
  - Yenileme bilgileri,
  - Garanti formları,
  - Arıza bildirim formları,
  - Gerekli kaynaklara erişim hakları,
  - Sorun gidermenin desteklenmesi,

- Çok sık karşılaşılan sorunlara yönelik çözümler,
  - Ürünlerin kullanım bilgileri,
  - Tüketicilerin birbirleri ile iletişimini sağlamak,
  - Diğer ürünlere ait bilgilerin sunumu,
  - Etkileşimli tüketici hizmetleri.
- Birlikte çalışılan şirketlere yönelik olarak;
    - Bağlantı bilgileri,
    - Gerekli çalışma prensiplerinin açıklanması,
    - Gerekli linkler ile tüketici hizmetlerinin gerekiyorsa birlikte gerçekleştirilmesi,
    - Gerekli yeniliklerin anında uygulanması,
    - Rakiplerden ve fırsatlardan haberdar olmak,
    - Fiyatlandırma politikalarında doğabilecek sorunları erken belirlemek.

## **2.7 Dünyada Elektronik Ticaret**

Dünyada elektronik ticaret konusundaki çalışmalarda, gelişmiş ülkeler başı çekmektedirler. Çeşitli ülkelerin Elektronik ticaret konusundaki çalışmaları incelenerek, G7 ülkeleri, Japonya, Avrupa Birliği, Avustralya gibi bazı ülkeler için aşağıda ana hatlarıyla özetlenmiştir (7).

### **2.7.1 G7 ülkelerinde elektronik ticaret konusunda yapılan çalışmalar**

G7 ülkelerinde elektronik ticaret ile ilgili çalışmalara, Amerika Birleşik Devletleri'nin öncülük yaptığı gözlenmektedir. ABD hükümeti, elektronik ticaretteki gelişmeleri yakından takip etmekte, bu konuda özel sektörün çalışmalarını desteklemekte ve hiçbir şekilde bir kısıtlama getirmemeye çalışmaktadır. Önümüzdeki birkaç yıl içinde bilgisayar ağları üzerinden yapılan ticaretin büyük miktarlarda olacağından konunun ciddiyeti artmaktadır.

G7 Pilot Projesi olarak sürmekte olan “Küçük ve Orta Büyüklükteki İşletmeler için Küresel Pazar” (Global Marketplace for SME's) kapsamında, elektronik ticaretin küçük ve orta boy işletmeler (KOBİ'ler) için yararlı ve yapılması gereken düzenlemeler, çalışma grupları oluşturarak ele alınmıştır.

Bu çalışma grupları:

- Enformasyon ağıları (information networks),
- Küreselleşme ve getirdikleri (globalization),
- Uygulama boyutu (deployment),
- Yasal düzenlemeler ve güvenlik (legal, regulatory, security),
- Uyum (interoperability),
- Finansman (financial issues),
- Pilot uygulamalar (testbeds and pilots)

konuları etrafında oluşmuş, bu konularda çerçeveler çizilmiş, hedefler belirtilmiş ve hedeflere uygun örgütlenmeler yapılmıştır.

G7 ülkeleri genel olarak kendi altyapılarında sürmekte olan elektronik ticareti yaygınlaştırmayı, vergi kayıplarını engellemek için kontrol mekanizmaları oluşturmayı, gereken güvenlik altyapısını, yasal düzenlemeleri ve yerel uygulamaları geliştirmeyi öngörmektedirler. Bu alanda yapılacak çalışmalar ile KOBİ'lerini uluslararası ticarete açmayı ve ticari işlemlerin bilgisayar ağları üzerinden yürütülmesini hedef seçmiş bulunmaktadır. Üzerinde çalıştıkları sistemler ile gümrük işlemleri de dahil olmak üzere ticari işlemlerin, güvenli ve hızlı bir şekilde bilgisayar ağları üzerinden yapılmasını sağlayacaklardır.

G7 organizasyonu, elektronik ticaretin internet üzerinden yapılacağı konusunu reddetmekte fakat ağ yapıları ve iletişim konusundaki gelişmelerden de konuyu soyutlamamaktadır. Bu gelişmelerden X.25, X.400, B-ISDN ve ATM ağ yapıları önemli görülmektedir. ATM yapıları ile daha hızlı iletişimin mümkün olması, çoklu ortam bilgilerinin internet yoluyla yayınlanmasını, dolayısıyla coğrafi olarak uzak yerler arasında internet üzerinden telekonferans ve görüntü transferini mümkün kılmaktadır. Böylece etkin ve yaygın ürün tanıtımları ile sunumlar, internet üzerinden yapılabilecektir.

G7 organizasyonu güvenlik ve yasal düzenlemeler konusunda, hem bilgilerin, hem de ticari işlemlerin güven altına alınması gerektiğini belirtmiş; ayrıca bu konudaki yerel ve uluslararası yasal düzenlemeler için bir model önermiştir. Güvenlik konusuna ek olarak, ticari ilişkide bulunacak tarafların güvenilirliklerinin tarafsız kurumlar tarafından onaylanmasının gerekliliği de belirtilmiştir.

### 2.7.2 Japonya’da elektronik ticaret konusunda yapılan çalışmalar

Japonya elektronik ticarete, oldukça temelden ve uzun vadeli yaklaşmaktadır. Japonya’da geliştirilmeye çalışılan yöntem politika ve teknolojilerin tümü, yakın bir gelecekte kağıt para yerine elektronik paranın geçmesini hedeflemektedir. Bu yüzden de Japonya için elektronik ticarete güvenliğin geliştirilmesi, bunun teknolojik ve yasal yönlerinin araştırılması en çok üzerinde durulan konuların başında gelmektedir. Bu ideale ulaşabilmek için gerekli yasal düzenlemelerin hazırlıkları da yine eCom bünyesindeki çalışma gruplarından “Onay Kurumları Çalışma Grubu” (Certification Authority Working Group) tarafından gerçekleştirilmektedir. Bu grup, eCom’a üye kurum ve kurumlara dağıtılmak üzere, elektronik ticarete yer alacak kurum ve kişilerin yetki ve sorumluluklarını ve birbirleriyle ilişkilerini belirleyen güvenli elektronik ticaret için bir ön yönetmeliği de içinde barındıran bir rehber yayımlanmıştır. Teknolojik olarak güvenli elektronik ticaret için ise, ISP (Information Security Protocols) adı altında, daha güvenli iletişimi sağlayacak protokol geliştirilmesi için çalışmalar yapılmaktadır.

### 2.7.3 Avrupa Birliği’nde elektronik ticaret konusunda yapılan çalışmalar

1980’lerin başından bu yana Avrupa Birliği (AB), Avrupa Ağı kapasitesini geliştirmek amacıyla AR-GE ağırlıklı programlar düzenlemekte (ESPRIT, RACE, ACTS’ vb.); bu kapsamda elektronik veri değişimi (EDI - Electronic Data Interchange) sistemlerine ve belirgin olarak da TEDIS (Trade EDI System) girişimine destek vermektedir. 1994’de Avrupa Komisyonu’nun isteği ve bilişim teknolojileri sektöründen önemli şirketlerin desteği ile Avrupa Bilişim altyapısını geliştirmek üzere 10 konuda hedef uygulama çalışmaları başlatılmıştır. Çalışmalardan dördü (KOBİ’ler için Telematik Servisleri, Elektronik Sunum, Avrupa Kamu Yönetimi Ağı ve Şehir Bilgi Ağı) elektronik ticaretle doğrudan ilişkilidir.

Güvenli, kağıtsız ticaret üzerine BOLERO projesi: sağlık, sosyal güvenlik, gümrük konularında ulusal ve AB aracı kurumları arasında idari doküman akışını sağlamaya yönelik IDA (Interchange of Data between Administration) programa önem verilmiştir.

#### 2.7.4 Avustralya'da elektronik ticaret konusunda yapılan çalışmalar

Dünyada bilgisayar dağılımında ikinci, kurulu bilgisayar gücü olarak sekizinci sırada yer alan Avustralya'da elektronik ticaretin gelişmesi hiçbir şekilde alışverişlerin parasal boyutlarıyla kısıtlanmamaktadır. Elektronik ticaret düzenlemeleri, sekiz kamu kurumu arasında dağıtılmıştır. Şu anda yürütülmekte olan ve "Yönetimsel Hizmetler Birimi" tarafından eşgüdümü sağlanan en önemli proje, bütün kamu tedarik hizmetlerinin elektronik ticaret ortamına geçirilişidir. Bu geçiş, Başsavcılık, Hazine ve Maliye Bakanlığı tarafından yürütülen politik ve yasal yapıların geliştirilmesi çalışmalarıyla desteklenmektedir. Avustralya hükümeti, endüstri ile işbirliği halinde ticaret ve taşıma zincirinde elektronik raporlama süreçlerinden sorumlu, Tradegate kuruluşunun finansmanını sağlamaktadır. Bu kuruluş sözleşmeli olarak ihracatçı ve ithalatçı şirketler ile Avustralya gümrüğü arasında elektronik bağlantıları kurmaktadır.

#### 2.8 Elektronik Ticaretin Güvenliği

Elektronik ticaretin yaygınlaşabilmesinin önündeki en önemli engel, açık iletişim ağları üzerindeki kullanıcıların birbirlerine kimliklerini kanıtlayabilmeleri sorunudur. Çözüm olarak önerilen açık anahtar altyapısı (public key infrastructure) kimlik kanıtlamaya ek olarak, bilgi bütünlüğü ve gizliliğini de sağlamaya yöneliktir. Açık anahtarlı (veya çift anahtarlı) şifreleme ve sayısal imzalama yöntemlerini kullanan her kişinin, gizli ve açık anahtar adları verilen ve birbirleriyle özel bir ilişkisi olan iki anahtar (sayı dizisi) vardır. Kişi, gizli anahtarını sadece kendisi bilir ve imza atarken ya da şifreli mesajlarını çözerken kullanır. Aynı kişinin açık anahtarı ise herkesin bilgisine ve kullanımına açıktır. Bu açık anahtar kullananlar ya anahtar sahibinin imzasını doğrulamak ya da yalnızca o açık anahtar sahibinin çözebileceği şifreli bir mesaj hazırlamak amacını güderler. Hangi açık anahtarın hangi kişiye ait olduğunu belgelemek için onay kurumları (Certification Authority) tarafından hazırlanan ve sayısal olarak imzalanan elektronik kimlik belgeleri (digital certificate) gereklidir.

Dünya pazarındaki en tanınmış onay kurumu olan VeriSign, RSA Bilgi Güvenliği, Ameritech ve Visa şirketlerinin ortaklığıyla kurulmuştur. Açık ağlar üzerinden birbirine ulaşmak isteyen iki kullanıcı, örneğin Çin'de, ABD'de veya Avustralya'da olabileceği için, kimliklerini karşılıklı olarak hızlıca kanıtlayabilmeleri, bütün onay kurumlarının uyum içinde çalışıyor

olmasına bağlıdır. Açık anahtar altyapısından amaçlanan da, eşgüdüm ve etkileşim içinde çalışması gereken onay kurumları/güvenilir üçüncü kuruluşlar/sayısal noterler/zaman damgası vurma veya anahtar bulma kurumları gibi kuruluşların sağladığı ulusal ve küresel hizmetler bütünüdür.

ABD'de açık anahtar altyapısı kurma sorumluluğunu üstlenen Ulusal Standartlar ve Teknoloji Geliştirme Enstitüsü, elektronik kimlik belgesi hazırlayan, içinde AT&T, Motorola, VeriSign gibi kuruluşların da bulunduğu 10 onay kurumunun yazılımlarını inceleyerek, bütün bu yazılımların uyum içinde çalışması için gerekli ölçütleri belirlemektedir.

Onay kurumlarının önemli bir bölümünü de banka ve benzeri kuruluşların oluşturması çok doğaldır. Örneğin, SET (Secure Electronic Transactions) standardını geliştiren MasterCard ve Visa, bu standardın gerektirdiği onay kurumu/sertifika otoritesi hizmetlerini de veren kuruluşlar olacaklardır.

### **2.8.1 Ülkelerin şifreleme politikalarındaki farklılıklar**

Dünya üzerindeki değişik ülkelerin şifreleme politikaları oldukça önemli farklılıklar göstermektedir. ABD, şifreleyici ürünlerin dış satımına kısıtlamalar koymakta, örneğin çok yaygın olarak kullanılan bir açık (çift) anahtarlı şifreleme yöntemi olan RSA algoritmasının anahtarı olan  $n$  sayısının 512 ikiliyi aşmasına izin vermemektedir. Ülke içinde 1024 ikili uygulamaları yasaklamadığı halde, bu durumdan endişe duymakta ve sade vatandaşlarla birlikte, yasadışı örgütlerin de çok önemli bir gizli iletişim gücü ele geçirmesini sakıncalı bulmaktadır. ABD'nin bu soruna bulduğu çözüm, Güvenilir Üçüncü Kuruluşlar yani GÜK'ler yardımıyla, herkesin gizli (özel) anahtarına, yasalar gerektirdiği zaman ulaşılmasını sağlayacak yöntemler geliştirilmesidir.

Japonya, şifrelemeye öncelikle ekonomiyi canlandırması açısından bakmakta, ulusal güvenlik yönünden şifrelemeyi bir tehdit olarak algılamamaktadır. Bu ülkelerden, GÜK'ler konusundaki politikası belli olmayan Japonya dışındakiler GÜK'lerin gerekliliğine inanmamakta, üçüncü kuruluşlara güvenilmesi gereksinimi yaratan bir toplumsal düzenlemenin GÜK'lerin güvenilirliğini zorlayacak yöntemler gelişmesine yol açabileceğini düşünmektedirler. Bu durum ise yasalara uyan vatandaşların haklarını zedeleyecek, sayısal

imzaların taklit edilebilmesine gizli mesajların istenmeyen kişiler tarafından okunmasına, ekonominin zarar görmesine yol açabilecek ve elektronik ticaretin gelişmesini engelleyebilecektir.

GÜK'leri gerekli veya gereksiz bulan iki ayrı görüş, onay kurumlarının gerekliliği konusunda ise birleşmektedir. Onay kurumlarını GÜK'lerden ayıran en önemli özellik, gizli anahtarları saklamaması ve anahtar çiftini üretip kullanıcıya verse bile kendisindeki gizli anahtar kopyasını imha etmesidir.

## 2.9 Türkiye'deki Durum

Türkiye'de birçok kamu kuruluşu, bilgisayar donanım ve yazılım altyapısını geliştirerek, yapmakla yükümlü olduğu işlerin otomasyonu, böylelikle insani hatalardan arındırılması, veri bankalarında toplanan bilgilerin erişim kolaylığı ve çabukluğundan yararlanılması gibi amaçlarla projeler yürütmektedir. İçişleri Bakanlığı Merkezi Nüfus İşleri Sistemi (MERMS), Sağlık Bakanlığı Temel Sağlık İstatistikleri Bilgi Sistemi, Gümrük Bakanlığı Gümrük Sistemleri Otomasyonu gibi projeler bunların başlıcalarıdır.

Elektronik ticarete başlangıç olarak düşünülebilecek bazı girişimler de vardır. Bilgisayar ürünleri ve kitap satan birkaç kurum ve büyük bir süper market, internet üzerinde hazırladıkları web siteleri ile kullanıcıya ulaşmakta, zengin ürün çeşitleri sergileyebilmektedirler. Fakat, bu uygulamaların hiçbirinde açık anahtarlı şifreleme kullanılmadığı için, kurumları kötü niyetli kullanıcıların aldatmasına karşı koruyabilecek bir önlem de yoktur. Diğer bir deyişle, kullanıcının iddia ettiği kimliğin kanıtlanması, kredi kartı numarasının kendisine ait olduğunun belirlenmesi ve mesajın yolda bozulmadığının gösterilmesine yarayacak sayısal imzanın olmaması, kurumu elektronik ticaret uygulamasında bir risk altına sokmaktadır.

Türkiye'de yaklaşık olarak evlerin yüzde 6-7 si, iş yerlerinin ise yüzde 11'inde bilgisayar bulunmaktadır. Bu oran gelişmiş ülkelerdeki oranlardan çok düşüktür. Bu nedenle, bilgisayar ağları üzerinden yapılacak elektronik ticarete, Türk halkının yaygın olarak katılacağı düşünülmemelidir. Öte yandan, az sayıda da olsa bazı ticari kuruluşlarımızın internet üzerinde web sayfaları hazırlayarak dünya ile iletişim kurabilmesi ve ürünlerini tanıtır dünya çapındaki

rekabetle baş edebilmesi mümkündür. Fakat, web üzerindeki satıcı sayısının bir çığ gibi büyüyeceği varsayılırsa, böyle yaygın bir rekabete uzun süre dayanılması kolay olmayacaktır. Bu nedenle, elektronik ticaretteki gelişmelerin uzun vadede küçük ve orta boy işletmelerden çok, büyük kurumlara yarayacağı öngörülebilir.

Açık ağlarda kimlik kanıtlama ve bilgi bütünlüğünü koruma sorunlarına çözüm getiren sayısal imza uygulaması, ağ üzerindeki kullanıcıların birbirlerine güven duymasına yarayacak; ama beraberinde yeni ve farklı bir güven sorunu getirecektir: Onay kurumlarına, onlar tarafından dağıtılan anahtar çiftlerinin düzgün üretilmiş olduğuna, kişilerin gizli anahtarlarının iyi, dürüst ve sağlam ellerde saklanacağına ya da üretilir üretilmez imha edileceğine, elektronik kimlik belgelerinin güncellenmesinin düzgün yürütüleceğine, yasal olarak gizli anahtara erişim durumlarında kişilik haklarının zarar görmeyeceğine, kısacası onay kurumun işleyişine güvenme veya güvenmeme sorunu.

Elektronik ticaretin yaygın ve hem kurumlara hem de müşterilere güven verecek şekilde kullanılmaya başlaması, güvenilir bir açık anahtar altyapısı (AAA) kurulmasıyla paralel olarak yürüyecektir. Bu altyapının kurulmasında öncülük görevi, dünyada olduğu gibi Türkiye’de de bankalara ve finans kurumlarına düşecek gibi görünmektedir. Elektronik ticaretin sağlıklı ilk uygulamaları, belirli bir bankada hesabı olan alıcı ve satıcılar arasında kolaylıkla gerçekleşebilir; çünkü aradaki bankanın varlığı, iki taraf için de gerekli güven ortamının oluşturacaktır.

İkinci aşamada, birden fazla bankanın kurduğu ortak yapıya kayıtlı alıcı ve satıcıların, internet üzerinden alışveriş yapabileceği güvenli bir ortam yaratılabilir. Bankalar arası Kredi Kartları Merkezi, böyle bir ortam için uygun bir zemin olacaktır. Fakat böyle bir ağ, internet üzerinde olsa bile, açık değil, kapalı bir bilgisayar ağı olacaktır; yani bu bankalar grubunda hesabı olmayan bir kişinin sisteme katılabilmesi mümkün olmayacaktır. Böyle bir yapının dünyaya bağlanabilmesi için gereken açık anahtarlı sayısal imza ve şifreleme yazılımı ya da donanımı dünya standartlarına uyum sağlamak zorundadır.

Gizli anahtarın korunabilmesi için en güvenli ortam, kişinin biyometrik bir özelliğine (göz, parmak izi gibi) bağlı olarak çalıştırılabilen akıllı kartlardır. Önümüzdeki yıllarda bilgisayarlara yüklenen sayısal imza ve şifreleme yazılımlarının, bilgisayar tarafından okunabilen akıllı kartlar sayesinde çalıştırılması beklenmektedir. Elektronik ticaret ve açık



anahtar altyapısıyla ilgili standartların oluřum s¼recine T¼rk bankalarının birlikte katılmaları bu konuda yapacakları yatırımları belirlenen standartlar doęrultusunda ve birbirleriyle eőg¼d¼m i¼inde y¼nlendirmeleri, genel verimlilięin artması a¼ısından ¼ok uygun olacaktır.



### 3 İLGİLİ ÇALIŞMALAR

Elektronik ticaret çok çeşitli teknolojileri temel almıştır. Bu teknolojilerden en önemlileri aşağıda verilmiştir (3).

- Kendi arasında işlenebilirlik (interoperability),
- Elektronik kataloglar, veritabanları ve çoklu-ortam sistemleri,
- Akıllı ajanlar,
- İş akışları ve süreç otomasyonu,
- Güvenlik Protokolleri.

#### 3.1 Kendi Arasında İşlenebilirlik

Elektronik ticaret çok çeşitli teknolojileri alt yapısında bulundurduğundan, bu teknolojilerin birbiri ile uyum sağlayacak şekilde birleştirilmesi büyük önem arz etmektedir. Böyle bir uyum, ancak belirli standartların oluşturulması ile sağlanacaktır. Elektronik ticaretin bu kadar önem kazanmasının bir nedeni de bu standartların hala oluşturulamamasıdır. Günümüz elektronik ticaret uygulamaları, Java programlama dili ile yazılmakta, ürün bilgilerinin alıcı-satıcı arasındaki değişimi ise XML ile sağlanmaktadır. Java, üretilen kodun; XML ise verinin sistemler arasında taşınabilmesini sağlamaktadır.

XML'in ne olduğunu tanımlayabilmemiz için önce Markup Language (İşaretleme Dili)'in ne anlama geldiğini açıklamakta fayda bulunmaktadır. Basılı yayının gelişmesiyle yayıncıların, yazılı metinlerin baskı makinelerinde hangi biçimde yayınlanması için hazırladıkları notlar ve özel semboller markup olarak ifade edilmekteydi. Bu, metnin belirli kısımlarının özel bir anlam kazandırmak üzere işaretlenmesi işlemidir. Bu amaçla kullanılan işaretler, kurallar ve gramer kümesi markup language (işaretleme dili) olarak tanımlanmaktadır. Bu tip işaretleme yapıları, bir çok ortamda görülebilmektedir. Örneğin ASCII kodlama standardının içerdiği bir çok kontrol karakteri veri iletişimi için kullanılmaktadır. Kelime işlem programları bir metnin içerdiği kısımları, yazı tiplerini, font ve stilleri ayırmak için metin içine gömülü bir çok işaretleri içermektedir. Programlama dilleri fonksiyonları, veri yapılarını, verileri ayırmak için bir takım semboller veya işaretler kullanmaktadır. Bu tip bir standart ayıraç, işaret veya

etiketler kümesi kullanmadan, taşınabilir ve paylaşılabilir bir uygulama geliştirilebilmesi çok güçtür.

Metin ve belgelerin kolay bir şekilde taşınabilmesi, paylaşılabilmesi ve işlenebilmesi için ilk işaretleme dili GML (Generalized Markup Language), 1960 sonlarında IBM'de yapılan araştırma çalışmaları sonunda ortaya çıktı. GML, daha sonra ANSI (American National Standard Institute) 1978'te oluşturulan bir grup tarafından geliştirilerek SGML (Standardized Generalized Markup Language) adı altında 1986 yılında ISO (the International Organization for Standardization) kurumunca uluslararası bir standart olarak kabul edildi. SGML, bir metin veya belge kümesinde kullanılan dilin gramer ve sözlük yapısını belirtmek için kullanılan bir dildir. Amerikan hükümeti kuruluşlarında, havacılık ve otomobil gibi büyük endüstri kurumları ve basın endüstrisinde bir belgeleme standardı olarak kullanılmaktadır; çok güçlü bir dil olmasına rağmen son derece karmaşık yapısı ve yüksek uygulama geliştirme maliyeti bu dilin yaygın bir şekilde kullanımını engellemiştir.

Tim Berners-Lee ve Anders Berlung 1989 yılında internet ortamında belge paylaşımını kolaylaştırmak için web uygulamalarının temel öğelerinden biri olan HTML (Hypertext Markup Language) dilini geliştirdiler. HTML, bir SGML uygulaması olarak geliştirildi. Diğer bir deyişle, HTML dilinin yapısı SGML dilinde tanımlandı. Çok basit yapısı ile son derece başarılı oldu. HTML dili, bir belgenin içerdiği başlık, font, resim ve tablo gibi bilgileri bilgisayar ortamında standart bir şekilde görüntülemek ve biçimlendirmek için geliştirilmiş bir dildir. Belgenin istenen şekilde sunulması tag (etiket) olarak ifade edilen işaretler ile sağlanmaktadır. Bu dilin geliştirilmesindeki temel amaç, belgeyi standart bir şekilde görüntülemektir. Bu dilin, yalnızca web tarayıcıları için sunum amaçlı olarak geliştirilmesi bugünkü web uygulamalarının gerektirdiği çok kısıtlamalar, XML dilinin geliştirilmesine yol açtı.

1996 yılında Word Wide Consortium (W3C, <http://www.w3.org>)'u SGML güç ve esnekliğini içerecek basit bir işaretleme dili oluşturmak amacıyla XML dilini tasarlamaya başladı. Şubat 1998'de XML 1.0 bir standart olarak W3C tarafından yayınlandı. XML dili SGML dilinin bir çok özelliğini içeren basitleştirilmiş dildir. SGML dilinin bir alt kümesidir. XML, SGML gibi bir meta dildir. Yani başka dillerin yapısını tanımlamakta kullanılan bir dildir (Akyokuş, 2000).

Microsoft ve Java dünyasında XML'in önemi giderek artmaktadır. Artık bir uygulama, XML içermiyorsa o uygulamaya büyük kurum uygulaması olarak bakılmamaktadır. Büyük ve kapsamlı bir uygulama yazmak için XML ile başlayan bir dizi standartlar serisinin öğrenilmesi gerekmektedir. Java'nın teknolojiye kazandırdığı taşınabilir kod gibi, XML de taşınabilir veriyi teknoloji dünyasına kazandırmıştır (Brett, 2000).

### 3.1.1 XML söz dizimi

XML belgeleri, işaretleme ifadeleri ve veri içeriğinden oluşan bir veri dosyası veya paketidir. İşaretleme ifadeleri; elemanlar (elements), varlık referansları (entity references), açıklamalar (comments), işleme komutları (processing instructions), CDATA bölümleri ve veri tipi tanımlamalarından (document type declarations) oluşur.

```
<?xml version="1.0" ?>
<!DOCTYPE e_posta SYSTEM "e-posta.dtd">

<e_posta>
  <kime>Herkes</kime>
  <kimden> Selim Akyokuş</kimden>
  <tarih> 7 Eylül 2000</tarih>
  <konu>XML ve XML Uygulamaları</konu>
  <mesaj> XML, HTML gibi etiket tabanlı bir dildir</mesaj>
</e_posta>
```

Şekil 3.1 XML örneği

XML'de geçen temel işaretleme ifadelerini detaylı incelemeden önce Şekil 3.1'deki örneği inceleyelim. Bu örnekte, XML dilinde tanımlanmış basit bir elektronik posta verilmiştir. Belgenin ilk satırı, `<?xml ... ?>` her zaman tanımlanması gereken XML işlem komutu bildirimidir. Bu, belgenin bir XML belgesi olduğunu ve hangi versiyonu kullandığını belirtmektedir. İkinci satır, seçimlik olarak tanımlanabilen belge tipi bildirimidir. Bu, belge için tanımlanmış olan DTD (veri tipi tanımlama) dosyasının isminin `e_posta.dtd` olduğunu ifade etmektedir. Bunların altındaki kısım XML belgesinin gövdesini oluşturmaktadır. XML belgesi gövdesi, hiyerarşik bir ağaç yapısında oluşmuş belge elemanlarından oluşur. Her eleman bir başlangıç ve bitiş etiketi içerir. Örneğin aşağıdaki *tarih* elemanı `<tarih>` başlangıç etiketinden, `7 Eylül 2000` karakter verisinden ve `</tarih>` bitiş etiketinden oluşmaktadır. Elemanlar diğer elemanları veya karakter verileri içerebilir. Bu örnekte `e_posta` elemanı;

*kime, kimden, tarih, konu* ve *mesaj* elemanlarını içermektedir. Bir belgenin en üst seviyesinde bulunan eleman, kök eleman olarak adlandırılır. Bu örnekte kök eleman, *e\_posta*'dır.

### 3.1.1.1 Elemanlar

Elemanlar (elements), XML dilinin en temel yapı taşlarıdır. Bir eleman, veri içeriğini kapsayan bir kutudur. Karakter verileri, diğer elemanları veya diğer işaretleme yapılarını içerebilir. Elemanlar, etiketlerle ayrılır. Her etiket, bir eleman tip ismi (element type name) içerir ve bir çift (“<”>”) açılı parantezle içe alınır. Eleman tip isimlerini oluşturmada olan kurallar, diğer programlama dillerindeki isim kurallarına benzemektedir. Her isim, bir harfle veya alt çizgi ( \_ ) ile başlayabilir. Tire (-) veya sayı ile başlayan isimler kullanılamaz.

Boş elemanlar (empty elements) dışındaki her elemanın, bir başlangıç etiketi bir de bitiş etiketi bulunur.

### 3.1.1.2 Başlangıç etiketleri

Bir elemanın başlama ayırıcı, başlangıç etiketi (start tag) olarak adlandırılır ve bir çift (“<”>”) açılı paranteziyle kapatılır. Aşağıda başlangıç etiket örnekleri verilmiştir.

<Kitap>  
<kitap>

Unutulmaması gereken bir diğer kural da, XML dilinin harflerinin büyük veya küçük olmasına duyarlı (case sensitive) olduğudur. Örnekteki <Kitap> ve <kitap> etiketleri aynı değildir.

### 3.1.1.3 Bitiş etiketleri

Bir elemanın bitiş ayırıcı, bitiş etiketi (end tag) olarak adlandırılır. Bitiş etiketi, eleman tip ismi öcesinde bir bölü (“/”) karakteri içerir ve açılı parantezleri ile kapatılır. Aşağıda bitiş etiketi örnekleri verilmiştir.

```
</kime>  
</kitap>
```

XML’de her bir başlangıç etiketi, bir bitiş etiketi ile mutlaka kapatılmalıdır. Başlangıç ve bitiş etiketi ile kapatılmış bir tam XML elemanı aşağıdaki gibidir.

```
<kitap> XML Unleashed </kitap>
```

### 3.1.1.4 Boş eleman etiketleri

Aşağıdaki örnekte ilk satırda olduğu gibi boş elemanlar (empty elements) herhangi bir içerik içermezler. Boş etiketler istenirse, etiketin sonuna tek bir bölü (“/”) işareti eklenerek kısaltılmış bir biçimde gösterilebilir. İkinci satırda bunun bir örneği verilmiştir.

```
<br></br>  
<br/>
```

### 3.1.1.5 Belge elemanı

XML belgeleri, mutlaka hiyerarşik etiket yapısında olmalıdır. Etiket hiyerarşisi içindeki en üst seviyedeki etiket, kök etiket (root tag) veya belge kökü (document root) olarak adlandırılır. Kök etiketin oluşturduğu eleman, belge elemanı (document element) olarak adlandırılır. Bu eleman, belgenin ağaç yapısındaki diğer bütün elemanları içine alır ve diğer bir elemanın içinde bulunamaz. Örneğin Şekil 3.1’de verilen elektronik posta örneğinde belge kökü e\_posta’dır.

Elemanlar iç içe yazıldığında, iç içe yazım kurallarına uyulmalıdır. Aksi halde bu, XML belgesinin hiyerarşik ağaç yapısını bozacaktır. Aşağıda yanlış iç içe yazım örneği verilmiştir.

```
HTML <B>
      <I> iç içe yazım kurallarına uymayı gerektirmez.
<B>                                     <I>
```

Aşağıda doğru iç içe yazım örneği verilmiştir.

```
XML,<B>
      <I> iç içe yazım kurallarına uymayı gerektirir. <I>
<B>
```

### 3.1.1.6 Özellikler

Elemanlara eklemek istediğimiz ek bilgiler, özellikler (attributes) olarak ifade edilir. Özellikler başlangıç etiketi içinde isim ve değer ikilileri olarak geçer. Aşağıda özellik örneği verilmiştir.

```
<kitap kitap_no="123456">
```

Buradaki kitap\_no, kitabın bir özelliğidir. Özellik değerleri mutlaka çift tırnak (") veya tek tırnak (') içine alınmalıdır.

### 3.1.1.7 Karakter veri

Bir elemanın başlangıç ve bitiş etiketleri içinde geçen metin içeriği karakter veri (character data) olarak adlandırılır. XML içinde özel amaçlı olarak kullanılan (&, <, >, ", ',) karakterleri olduğu gibi, bir karakter veri içinde geçemezler. Bu karakterler, işaretleme etiketleri içinde özel amaçlarla kullanılan ayıraçlardır.

XML'de HTML'den farklı olarak, etiketler içine yazılmış olan veri içinde geçen boşluklar (white space) korunur ve uygulamalara iletilir. Etiket tanımları içinde veya özellik tanımı içinde geçen boşluklar ise HTML'de olduğu gibi çözümleyici tarafından yok edilir.

### 3.1.1.8 Karakter ve varlık referansları

SGML ve HTML gibi ASCII karakter seti içinde karakterleri göstermek için XML, iki yöntem kullanmaktadır. Birinci yöntem, karakter referanslarıdır (character references). Karakter referansları gösterilecek karakteri onluk bir sayının başına “&#”, onaltılık bir sayının başına ise “&#x” ekleyerek gösterme imkanı sağlar. Örneğin copyright karakteri, ©, ondalık olarak “&#169” veya onaltılık olarak “&#xA9” şeklinde XML’de gösterilebilir.

Varlık referansları (entity references), verilecek bir isim vasıtasıyla istenilen karakterleri göstermek için kullanılır. Varlık referansları bir (&) karakteri, bir isim ve noktalı virgülden (;) oluşur. Örneğin XML’de özel olarak kullanılan (&, <, >, ', ") karakterleri (&amp; &lt; &gt; &apos; &quot;) varlık referansları ile gösterilir. Bu referanslar, XML’de daha önce tanımlanmış standart referanslardır. Bu referanslar haricinde özel olarak kullanılmak istenilen varlık referansları DTD içinde tanımlanmalıdır.

### 3.1.1.9 Açıklamalar

Açıklamalar (comments) HTML’de olduğu gibi “<!--” ile başlar ve “-->” ile biter. Açıklamalar bir XML belgesinin herhangi bir kısmında geçebilir ve belgenin bir parçası sayılmazlar.

### 3.1.1.10 İşleme komutları

XML belgelerini kullanan uygulamalara her hangi bir komut geçilmesi istenildiğinde bu, işleme komutları (processing instructions) ile yapılır. İşleme komut yapısı aşağıda verilmiştir.

```
<?hedef ...komutlar ... ?>
```

İşleme komutları, XML işlemcisi tarafından uygulamaya iletilir ve açıklamalar gibi bir XML belgesinin parçası sayılmazlar.



### 3.1.1.11 CDATA bölümleri

CDATA bölümleri (CDATA sections), XML’de kullanılan işaretleme ifadelerinin göz ardı edilmesi gereken metinleri yazmada kullanılır. XML çözümlenmesi, bu bölümde metin içinde yer alan işaretleme ayraçlarını ve etiketlerini normal karakter olarak algılar. Bir CDATA bölümü yapısı aşağıda verilmiştir.

```
<![CDATA[ ... ]]>
```

Bu yapıda “...” olarak ifade edilen kısım, “[” hariç, herhangi bir metni içerebilir. Aşağıda CDATA bölümü örneği verilmiştir.

```
<![CDATA[
  *P = &A
  C = (i <= 5);
]]>
```

### 3.1.1.12 İyi oluşturulmuş ve doğru belgeler

XML söz dizimi (syntax) kurallarına uyan XML belgeler, iyi-oluşturulmuş (well-formed) belgeler olarak adlandırılır. Eğer belge için tanımlanmış olan DTD’de geçen gramer kurallarına uyuyorsa, bu durumdaki belgeler de doğru (valid) belgeler olarak adlandırılır.

### 3.1.1.13 Belge tipi bildirimi

Belge tipi bildirimleri (DTD), bir XML belgesinin giriş kısmında tanımlanan belgenin gramer tanım kurallarını belirleyen hangi harici ve dahili DTD’yi kullandığını belirtir. Bu kısım, varlık referansı tanımlamaları gibi ek bilgiler de içerebilir.

```
<!DOCTYPE kök_eleman_ismi SYSTEM "sistem_tanımlayıcısı">
<!DOCTYPE kök_eleman_ismi PUBLIC
  "public_tanımlayıcısı""sistem_tanımlayıcısı">
```

Şekil 3.2 Harici belge tipi bildirimi

Harici belge tipi bildirim Şekil 3.2’de verilmiştir. Bu tanımdaki kök\_eleman\_ismi, belge DTD’sinde en üst seviyede tanımlanan kök elemanını belirtir. Sistem\_tanımlayıcısı ise harici olarak kullanılan DTD’de URI (Uniform Resource Identifier) bir referansı içerir. İkinci satırdaki formatta XML çözümleyici, ilk önce public\_tanımlayıcısı’nı kullanarak bir URI referansı oluşturmaya çalışır. Eğer bu mümkün olmazsa, sistem\_tanımlayıcısı’ndaki referansı kullanır. Şekil 3.3’te adres belgesinin belge tipi bildirim kısmı gösterilmiştir.

```
<?xml version="1.0"?>
<!DOCTYPE addressbook SYSTEM "http://xyz.com/AddressBook.dtd" [
<!ENTITY amp "&#38;#38;">
<!ENTITY apos "&#39;">
]>
```

Şekil 3.3 Adres belgesinin DTD’si

İlk satır, adres belgesindeki kök elemanın addressbook olduğunu belirtmektedir. Belgenin DTD’si “<http://xyz.com/AddressBook.dtd>” dosyasında saklanır. Bunun dışında köşeli parantez içinde tanımlanan kısım ise varlık referanslarını tanımlamaktadır. Bu kısımda istenirse, dahili belge tipi tanımlamaları da bulunabilir.

DTD, dokümanı ağaç yapısı şeklinde tanımlar. DTD’nin amacı dokümandaki hiyerarşiyi tanımlamaktır. DTD aşağıdaki bilgileri içerir (Yılmaz, 1999):

- Elemanların adlarını,
- Her elemanın içeriğini,
- Kaç defa tekrarlandığını ve sırasını,
- Son etiketlerin başlangıçlarının gösterilip gösterilmeyeceğini,
- Özelliklerin içeriklerini ve varsayılan değerlerini,
- Birimlerin adlarını.

Bir XML belgesinin yapısı, DTD’de Çizelge 3.1’de verilen işaretleme ifadeleri ile tanımlanır.

Çizelge 3.1 DTD işaretleme ifadeleri

DTD ifadesi	Anlamı
ELEMENT	Bir XML elemanını tanımlar.
ATTLIST	Bir eleman tipinin alabileceği özellikleri tanımlar.
ENTITY	Belli bir içeriğin bir isme bağlanarak, o isimle ilişkilendirilmesini sağlar.
NOTATION	XML, harici içeriğin tanımlanmasını sağlar. Bu tanımlanan içerik, XML belgesinin bir parçası olarak çözümlenmez.

### 3.1.1.13.1 Elemanlar

Elemanlar, XML belgesinin en temel yapısıdır ve ELEMENT etiketi ile tanımlanır. Dört çeşit eleman tipi vardır:

- Boş (Empty): Eleman herhangi bir içerik içermez. Elemanın özellikleri olabilir.
- Yalnızca eleman (Element Only): Eleman yalnızca alt seviye (child) elemanları içerebilir.
- Karışık (Mixed): Eleman alt seviye elemanlar ve karakter veri karışımını içerebilir.
- Any: Eleman DTD'nin izin verdiği içeriği içerebilir.

#### 3.1.1.13.1.1 Boş elemanlar

Boş elemanlar aşağıdaki gibi tanımlanır.

```
<!ELEMENT Elemanİsmi EMPTY>
```

Boş eleman örneği aşağıda verilmiştir.

```
<!ELEMENT img EMPTY>
```

Boş eleman, XML belgesi içinde daha önce açıklandığı gibi, eğer bir özelliği yoksa aşağıdaki ilk iki satırdaki gibi geçecektir.

```
<img/>  
<img></img>
```

### 3.1.1.13.1.2 Yalnızca eleman içeren elemanlar

Bu tipteki elemanlar aşağıdaki gibi tanımlanır.

```
<!ELEMENT Elemanİsmi İçerik-Modeli>
```

Bu yapıda, İçerik-Modeli yalnızca bu elemanın altında kalacak alt elemanları içerebilir. Örneğin Şekil 3.1'de verilen elektronik posta kök elemanı e-posta'nın tanımı aşağıdaki gibidir.

```
<!ELEMENT e-posta kime kimden tarih? konu? mesaj>
```

Bu örnek e-posta elemanı *kime*, *kimden* ve *mesaj* alt elemanlarını mutlaka gösterilen sırada XML belgesi içinde geçmesi gerektiğini göstermektedir. *Tarih* ve *konu* elemanları ise seçimlidir. Bu elemanların seçmeli olduğunu (?) sembolü göstermektedir. DTD'de çeşitli amaçlarla kullanılan sembollerin anlamları Çizelge 3.2'de listelenmiştir:

Çizelge 3.2 DTD sembol anlamları

Eleman Tanımı	Anlamı
A?	Seçimlik; A alt elemanı bir defa veya hiç gözükmeyebilir.
A+	A alt elemanı, mutlaka bir defa veya daha fazla gözükür.
A*	A alt elemanı, sıfır veya daha fazla gözükabilir.
A,B	B elemanı, A elemanından sonra gözükmelidir.
A   B	A veya B'den bir tanesi gözükabilir.
()	Parantez içinde geçecek elemanları guruplar. Örneğin (A, B)+ ifadesi A ve B'nin ardışık olarak bir veya daha fazla geçeceğini göstermektedir.

### 3.1.1.13.1.3 Karışık elemanlar

Karışık elemanlar, hem karakter veri hem de alt elemanları içerebilirler. Karakter veri #PCDATA terimi ile gösterilir ve parsed karakter veri anlamındadır.

```
<!ELEMENT konu (#PCDATA)>
<!ELEMENT not (#PCDATA | tarih | konu)*>
```

Şekil 3.4 Karışık elemanın DTD'de tanımı

Şekil 3.4'te karışık eleman örnekleri verilmiştir. İlk satırdaki ifade, *konu* elemanının yalnızca karakter veri içereceğini göstermektedir. Bu tip, yalnızca veri içeren eleman tanımlarında #PCDATA parantez içine alınmalıdır. İkinci satırdaki karmaşık *not* elemanın karakter veri, tarih veya konu elemanlarından oluşabileceğini göstermektedir. Şekil 3.4'te verilen DTD tanımı, Şekil 3.5'te verilen XML dokümanını tanımlar.

```
<not>
  <tarih> 1 Ekim 2000 </tarih>
  Okul kayıtları başlayacak.
</not>
```

Şekil 3.5 Karışık elemanın XML ile gösterimi

### 3.1.1.13.1.4 ANY elemanları

ANY tanımı aşağıda verilmiştir.

<!ELEMENT Elemanİsmi ANY>

ANY elemanları herhangi bir yapıya sahip değildir. ANY elemanı karakter veri, tanımlanmış elemanlar veya bunların bir karışımını içerebilir. ANY elemanları her hangi bir yapı belirtmediği için kullanılması tavsiye edilmemektedir.

### 3.1.1.13.2 Özellikler

Daha önce de belirtildiği gibi özellikler elemanlar hakkında ek bilgi tanımlamak için kullanılır. Bir eleman hakkındaki özellik bilgileri, aşağıdaki gibi tanımlanır.

<!ATTLIST Elemanİsmi Özellikİsmi ÖzellikTipi *Default*>

Bir özellik bir isim ve tip bilgisini içerir. Default bir değer olabilir veya özellik kullanımı hakkında bilgi verir. Dört çeşit default değer tipi, Çizelge 3.3'de verilmiştir.

Çizelge 3.3 DTD default değer tipleri

Default	Anlamı
#REQUIRED	Özellik her zaman tanımlanmalıdır.
#IMPLIED	Özellik seçimlidir.
#FIXED <i>Değer</i>	Özellik, sabit bir <i>Değer</i> 'e sahiptir. Özellik, eleman içinde tanımlı olmazsa, çözümleyici bu sabit <i>değer</i> 'in olduğunu varsayar.
Değer	Özellik eleman içinde tanımlı olmazsa, çözümleyici bu <i>değer</i> 'in olduğunu varsayar. Eğer tanımlı ise özellik farklı bir değer içerebilir.

Örneğin aşağıdaki örnekte, Kitap elemanının BasımTarihi diye bir özelliğinin olduğu ve bu özelliğin karakter veri olduğu (CDATA) belirtilmektedir.

```
<!ATTLIST Kitap BasımTarihi CDATA #REQUIRED>
```

Bu özellik, bir XML belgesi içinde şu şekilde geçebilir:

```
<Kitap BasımTarihi="1.1.2000"> ..... </Kitap>
```

Aşağıdaki örnek, *deniz* elemanının *renk* özelliğinin sabit olarak mavi değerine sahip olduğunu göstermektedir.

```
<!ATTLIST deniz renk #FIXED "mavi">
```

DTD değişik özellik tiplerini desteklemektedir. Bu özellik tipleri, Çizelge 3.4'te verilmiştir.

Çizelge 3.4 DTD özellik tipleri

Özellik Tipi	Anlamı
CDATA	Karakter veri (Unparsed).
Enumerated	Özelliğinin sahip olabileceği değerler kümesi.
NOTATION	DTD'de tanımlanmış herhangi bir notation.
ENTITY	Harici bir varlık.
ENTITIES	Birden fazla boşlukla (white space) ayrılmış varlıklar.
ID	Bir belge içinde geçebilecek tek tanımlayıcı bir değer (unique identifier).
IDREF	DTD'de tanımlanmış olan, bir ID'ye olan referans.
IDREFS	Boşlukla (white space) ayrılmış ID'ler serisine referans.
NMTOKEN	XML token karakterlerinden (harf, sayı, nokta, tire, iki nokta, alt çizgi) oluşan bir isim.
NMTOKENS	XML token karakterlerinden oluşan, boşlukla ayrılmış bir çok isim.

Enumerated özellikler, bir dizi değer kümesi içerir. Özellik, bunlardan bir değeri alabilir. Aşağıdaki örnek *kalem* elemanının *renk* özelliğinin *mavi*, *kırmızı*, *yeşil* veya *beyaz* olabileceğini göstermektedir. Eğer bir renk seçilmezse default olarak *kırmızı* renk alınır.

```
<!ATTLIST kalem renk (mavi | kırmızı | yeşil | beyaz) "kırmızı">
```

Aşağıdaki örnek, DTD’de tanımlanmış olan notation’ları özellik olarak kullanmaktadır. Notation, XML harici veri tiplerini tanımlamakta kullanılır. Bu örnek resim formatının *gif* veya *jpeg* olabileceğini belirtmektedir.

```
<!ATTLIST resim format NOTATION (gif | jpeg) #REQUIRED>
```

ID tipindeki özellik değerleri bir belge içinde yalnızca bir defa geçebilir. Bu bize bir elemanı, tek olarak tanımlamamızı sağlar. Örneğin, personel diye bir elemanımız olsun. Her personel, personel-no adlı bir özellik içersin. Her personelin numarası farklı olacağından bu özellik ID olarak tanımlanır.

```
<!ATTLIST personel personel-no ID #REQUIRED>
```

IDREF, ID tipindeki elemanlara referans vermemizi sağlar. Örneğin mühendis adlı bir elemanımız olsun. Mühendis ile personel arasındaki bire bir ilişkiyi IDREF ile Şekil 3.6’daki gibi tanımlayabiliriz.

```
<!ELEMENT mühendis EMPTY>
<!ATTLIST mühendis per-no IDREF #REQUIRED>
```

Şekil 3.6 IDREF’in DTD’de tanımı

Bu, mühendisin personel bilgilerini personel elemanından elde etmemizi sağlar. Şekil 3.7’de bir XML belgesi örneği verilmiştir.

```
<personel personel-no="12345">
  <isim> Ali Veli </isim>
  ....
</personel>
....
<mühendis per-no= "12345"/>
```

Şekil 3.7 IDREF’in XML ile gösterimi



IDREFS, iki eleman arasındaki bire çoklu ilişkileri göstermekte kullanılabilir. Örneğin proje-grubu adlı bir elemanımız olsun. Bir proje grubu bir çok personel içerecektir. Bu ilişki, Şekil 3.8’de gösterilmiştir.

```
<!ELEMENT proje-grubu EMPTY>
<!ATTLIST proje-grubu per-no IDREFS #REQUIRED>
```

Şekil 3.8 IDREFS’in DTD’de tanımı

Bu yapıdaki bir XML belgesi Şekil 3.9’da verilmiştir. Bu belge 12345, 67891, 23456 numaralı personelin proje gurubunda yer aldığını belirtmektedir.

```
<proje-gurubu per-no= "12345 67891 23456">
```

Şekil 3.9 IDREFS’in XML’de gösterimi

Özellik değerlerinin, token olarak varsayılması gereken durumlarda NMTOKEN veya NMTOKENS tipi kullanılır. Token, boşluk içermeyen bir değerdir. XML çözümleyici, bir özelliği token olarak işlerken, aradaki boşlukları (white space) tek boşluğa indirger ve baş ve sondaki boşlukları siler. Bu işlem, özellik değeri normalizasyonu (attribute value normalization) olarak adlandırılır. Şekil 3.10’da bir çok token içeren bir NMTOKENS özellik tipi tanımı ve XML ifadesi verilmiştir.

```
<!ATTLIST personel eğitim NMTOKENS #IMPLIED>
<personel eğitim="ilkokul ortaokul lise "
```

Şekil 3.10 NMTOKENS’in DTD tanımı ve XML gösterimi

### 3.1.1.13.3 Varlıklar

Varlıklar herhangi bir veri parçasına bir isim vererek, bu veri parçalarına referans vermemizi sağlar. Bunu C dilindeki DEFINE fonksiyonu gibi düşünebiliriz. Varlık tanımlamaları bir defa tanımladığımız bir değeri bir çok yerde ismi ile referans vererek tekrar kullanabiliriz. Üç çeşit varlık tanımlaması vardır:

- Genel Varlıklar (General Entities)
- Parametre Varlıklar (Parameter Entities)
- Karakter Varlıklar (Character Entities)

### 3.1.1.13.3.1 Genel varlıklar

Genel varlıklar, bir metin parçasına bir isim vererek bu isimle metni kullanmamızı sağlar. Örneğin aşağıda bir şirket ismi için yapılmış tanımlama gösterilmiştir.

```
<!ENTITY şirket "XYZ Bilgisayar Limited Şirketi A.Ş.">
```

Tanımlamayı aşağıdaki gibi ismin başına (&) harfi ve sonuna (;) ekleyerek kullanabiliriz.

```
<reklam> Firmamız &şirket; internet danışmanlığı hizmeti sunar. </reklam>
```

Bu ifadedeki &şirket referansı tanımlamadaki metin ile yer değiştirecektir. Genel varlık tanımlamalarını, harici bir dosyaya ulaşmak için de kullanabiliriz. Örneğin aşağıdaki tanımlama, *readme.txt* dosyasına referans tanımlamaktadır.

```
<!ENTITY aciklama SYSTEM "http://www.xyz.com/readme.txt">
```

### 3.1.1.13.3.2 Parametre varlıklar

Genel varlık tanımlamaları, XML belgesi içinde bir metin için tanımladığımız isim ile belgenin içinde yer almasını sağlar. Parametre varlık tanımlamaları aynı işlevi, DTD'nin içinde yapmamızı sağlar. Parametre varlıkları aşağıdaki gibi tanımlanır.

```
<!ENTITY % Varlıkİsmi VarlıkTanımı>
```

Aşağıdaki örnekte, boyut isimli bir parametre tanımı yapılmıştır.

```
<!ENTITY % boyut "uzunluk genişlik yükseklik">
```

Parametre varlıkları, varlık ismi başına (%) sonuna (;) konarak DTD içinde referans edilir. Aşağıda boyut'a yapılmış olan referans verilmiştir.

```
<!ELEMENT duvar (%boyut);>
```

### 3.1.1.13.3 Karakter varlıklar

Karakter varlık referansları, verilecek bir isim vasıtasıyla istenilen karakterleri göstermek için kullanılır. Karakter varlık referans tanımlamaları, aşağıdaki örnekte olduğu gibi yapılır:

```
<!ENTITY copyright "&#169" >
```

Bu tanımlamada ASCII kodlamada, 169, copyright karakterini (©) belirtmektedir. Karakter varlık referansları bir ampersand (&) karakteri, bir isim ve noktalı virgülden (;) oluşur. XML metninde *&copyright;* yazdığımız yerleri, © sembolü ile yer değiştirecektir.

### 3.1.1.13.4 Gösterim

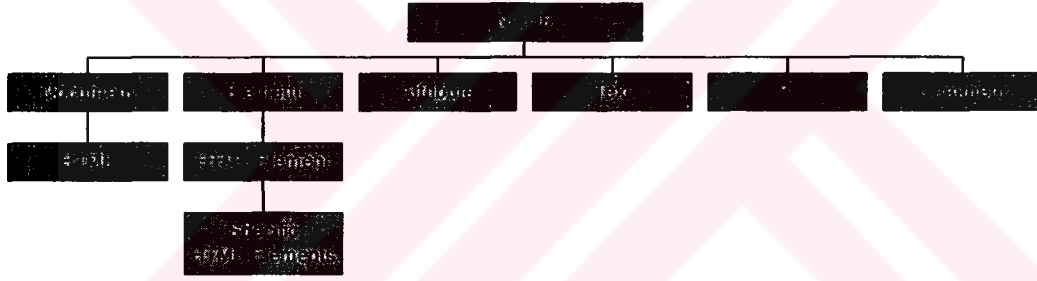
Gösterim tanımlamaları, XML harici verinin ve bu harici ile ilişkilendirilecek bir yardımcı uygulamanın (helper application) tanımlanmasını sağlar. XML çözümleyici, XML verisi üzerinde çözümüleme işlemini yapar. XML harici verilerin işlenmesi, örneğin *gif* şeklinde tanımlanmış bir resim verisi, ancak bu veri şekilleri ile ilişkilendirilecek bir uygulama ile yapılabilir.

```
<!NOTATION jpg SYSTEM "jpgviewer.exe">  
<!NOTATION gif SYSTEM "gifviewer.exe">
```

Yukarıdaki örnek, *jpg* ve *gif* şeklindeki tanımlanmış verileri ilgili uygulamalar ile ilişkilendirmektedir. Bu notation, jpg dosyalarını jpgviewer.exe, gif dosyalarının gifviewer.exe uygulaması ile işlenmesi gerektiğini belirtmektedir.

### 3.1.2 Document Object Model (DOM)

DOM, platform-dil-nötr arayüzdür. Programların ve scriptlerin dokümanın stilini, yapısını, içeriğini dinamik olarak değiştirmeyi ve erişimi sağlar (4). DOM bir dokümanı, nesne hiyerarşisi olarak tanımlar. Bu, node olarak adlandırılır. Node, DOM'da temel nesnedir. Bu nesneye, bağlı diğer nesnelere olabilir. Element nesnesi HTML ve XML'deki elemanları belirler. Şekil 3.11'de nesne yapısı verilmiştir.



Şekil 3.11 DOM nesne yapısı

### 3.2 Elektronik Kataloglar, Veritabanları ve Çoklu-Ortam Sistemleri

Elektronik ticarete önemli olan diğer bir konu ise; ürün tanımlarının hızlı ve kolay şekilde alıcıya sunulmasıdır. Arama mekanizmaları sayesinde ürünün bilgileri ve hangi internet adresinde olduğu elde edilmektedir. Fakat, ürünün internet üzerinden satın alınması için sadece ürün hakkında bilgi yeterli olmamaktadır. Ürün hakkındaki bilgiler alıcıya sunulurken; alıcının, bilgileri istediği özelliğe göre gruplamasına ve karşılaştırmasına izin veren standartlaşma sağlanmalıdır. Daha önemlisi alıcı, herhangi bir ürünü tamamlayan diğer ürünlerin ve servislerin neler olduğu hakkında bilgilendirilmelidir.

Yukarıda tanımlanan amaçlar, web-tabanlı elektronik kataloglar sayesinde elde edilecektir. Fakat, geliştirme, test etme, teknolojileri uyarlama, standartlar, servisler ve iş uygulamaları alanlarında, güvenilir elektronik katalog değişimini sağlayan, çok fazla deneyin yapılması gerekmektedir.

Elektronik kataloglar birbirleriyle uyumlu çalışmalıdır. Bunun için elektronik katalog haberleşme protokolü geliştirilmelidir. Bu protokol, internet üzerindeki elektronik katalogları kontrol eden, yapısallaştıran, ürün bilgilerini sunan, yöneten çok çeşitli veritabanlarını birbirine bağlayan bir köprü olacaktır. Bu protokolün daha da geliştirilmesi ile kataloglar, sistem platformundan, işletim sisteminden ve veri şekliinden bağımsız bir şekilde cevap verebileceklerdir. Sonuç olarak elektronik kataloglar (2):

- Ürün karşılaştırması sunmalıdır.
- Büyük kurumlar için satıcılar özel indirimler uygulamaktadır. Elektronik kataloglar normal alıcılar için ayrı fiyat, özel alıcılar için ayrı fiyat tanımlamasını sağlamalıdır.
- Ürünü tamamlayan diğer ürünleri tanımlamaya izin vermelidir. A malı alınırsa, B malının ve C malının alınması gerektiği bilgisi sunulmalıdır.
- Oluşturulacak standart sorgular sayesinde satıcılar alıcılara üç farklı yolla tamamlayıcı ürünleri bulabilmeyi sağlarlar. Birincisi, kodlanmış standart sorguları tüketici, alıcıya iletilir. Satıcı da üreticiye, arbulucuya ve dağıtımçıya, alıcı hangisini seçmişse, gönderir. İkincisi, web sayfasına, asıl ürünü sağlayan tarafından belirlenen, tamamlayıcı ürünlere bağlantılar gösterilir. Üçüncüsü ise satıcıların, sadece kendi ürünlerinin olmadığı, aynı zamanda tamamlayıcı ürünlerin de olduğu bir elektronik katalog yaratmasıdır.
- Elektronik kataloglar birbiriyle uyumlu çalışmalıdır. Bunun için ürün tanımlamaları standartlaştırılmalı, kataloglar arası haberleşmeyi sağlayan endüstri standart protokolü oluşturulmalı, protokol oluşturulduktan sonra ürün hakkında ek bilgilerin, fotoğrafların, videoların bağlantılar halinde gösterildiği alıcı sorguları oluşturulmalıdır.

### 3.3 Akıllı Ajanlar

Akıllı ajanlar, bir kullanıcının yapmak istediklerini kullanıcı adına yapan yazılımdır. Ajanlar yazılım olmalarına rağmen, diğer yazılımlardan farklıdır çünkü, ajanlar karmaşık işlemleri dışarıdan bir müdahale olmadan yerine getirebilirler. Oysa birçok yazılım dışarıdan müdahalelerle işlemleri yerine getirmektedir. Bir ajan aşağıdaki özelliklere sahip olmalıdır:

- **Bağımsızlık:** Ajanlar kullanıcıdan veya başkasından doğrudan girdi almadan işlem yaparlar. Kendi hareketleri ve durumları üzerinde kontrol yetenekleri vardır. Yani ajanların işlemlerini yapabilmeleri için, diğer yazılım paketlerinde olduğu gibi dış ortamdan müdahaleye gerek yoktur. Ayrıca ajanlar, kendi hareket tarzları ve içinde buldukları durum üzerinde kontrol yetkisine sahiptir.
- **Muhabere yeteneği:** Ajanlar, diğer ajanlarla ve kullanıcılarla ajan dili (örneğin KQML) ile haberleşirler.
- **Tekrar etkin hale gelme yeteneği:** Ajanlar çevresindeki değişiklikleri algılar ve zamanında değişikliklere cevap verirler.
- **Görevini Yürütme:** Ajanlar sadece çevresindeki değişikliklere cevap vermek için harekete geçmez, kendi kendini harekete geçirerek varoluşunun gereklerini de yerine getirir. İşlem yapmak için daima çevreden bir dürtü beklemezler.
- **Devamlılık:** Ajanlar durmadan çalışan programlardır, sadece bir defa çalışıp sonlanmazlar.
- **Amaca bağlılık:** Ajanlar işlerin karmaşıklığından etkilenmeden amaçlarını gerçekleştirirler.

O halde, ajanlar yapmanız gereken işi, zamanınız olmadığı için sizin yerinize, yapmanız gereken şekilde yapan yazılımlardır.

Elektronik ticaretin çoklu ajan yapısını sağlayan temel yapıları, aşağıda belirtilmiştir:

- Her tarafa yayılan middleware (pervasive middleware), örneğin internet, HTTP, web-nesneleri, HP E'Speak, Microsoft BizTalk ve Sun's Jini.
- B2B haberleşme standartları ve modelleri (XML ve CommerceNet tarafından yazılan eCo, CommerceOne tarafından yazılan CBL, ontology.org, RosettaNet tarafından geliştirilen terimler).
- Akıllı Ajanlar (genetik algoritmalar, bulanık mantık, yapay-sinir ağları, kural-tabanlı sistemler ve ajan teknolojisi).

Ajan teknolojileri, çok-ajanlı-elektronik ticaret sistemlerini kurmak için gerekmektedir. Düzgün yazılmış XML-tabanlı ajan iletişim dilleri, terimler, ajanları gruplamak için modelleme teknikleri, iş akış metotları ve teknolojileri ajan koreografilerinde önemli gelişmeler sağlayacaktır. http; XML, ajanlar, KQML ve iş akışları uyumlu yazılım parçası teknolojisini sağlayacaktır. Böylelikle çok kısa sürede yeni uygulamalar yazılabilecektir. XML/http-tabanlı eCo, BizTalk özellikleri haberleşmesi ve terim altyapısı, geleceğin nasıl olacağını bize gösterecektir (GRISS, 2000).

### 3.4 İş Akışları ve Süreç Otomasyonu

İş akışı, bir iş sürecini başarıyla tamamlamak için görevler veya prosedürsel adımlar, insanlar, araçlar topluluğudur. Örneğin, bankadan ödünç para almak için sizin talep formu doldurmakla başladığımız, bankanın bu işlem için belirlediği kurallar dahilinde, belirli görevlilerine takip ettirdiği bu ödünç para başvurusunun, ödünç alma işlemi ile sonuçlanmasına kadar bu işe katılan tüm birimler topluluğu ve yapılan işlemler ödünç para alma iş akışını oluşturur. İş akış süreci, aktivitelerin koordinasyonu, kontrolü ve haberleşmesini sağlar. İş akış uygulaması ise, iş akışına katılanlar arasında daha iyi haberleşme ve işbirliği sağlayarak iş sürecini otomatikleştiren bir yazılımdır (Arpınar v.d., 2000).

Elektronik ticaret uygulamalarında bir iş süreci, seri adımlar dizisi olarak modellenebilir. Adımlar arasında veri ve kontrol akış bağımlılıkları bulunmaktadır. Bu yüzden, iş akışları elektronik ticarete kullanılır ve elektronik ticaret, iş süreçlerinin yönetimini sağlar.

İyi bir iş akış uygulaması; iş akış ilerlemelerini kontrol eden kuralları koymayı, özel görevlerin verildiği kullanıcıların rollerini belirlemeyi, bir adımdan diğer adıma belirli bir süreci tanımlı yol üzerinden yönlendirmeyi sağlar. Bir iş akış uygulaması, genelde şu elemanlardan oluşan bir yapı içerir:

- **İş Akış Kontrol İşlemcisi:** Bu işlemci, çalışan iş akışlarını takip etmekle, tek bir iş akış adımının yürütülmesini kontrol etmekle, kullanıcıdan ve kullanıcıya iş akış metotlarını göndermekle ve almakla görevli merkezi elemandır.
- **Mesajlaşma Mekanizması:** İş Akış Kontrol İşlemcisi ile iş akış kullanıcıları arasında haberleşmeyi sağlar. Kullanıcı, kabul/red şeklinde isteklere cevap gönderir.
- **İş Akış İstemcisi:** Kullanıcılar iş akış isteklerini aldıklarında ve cevaplarını iş akış makinesine gönderdiklerinde, süreci yürütmek için istemci yazılımcısı kullanırlar. İstemci yazılımcısı; iş akış kontrol işlemcisi tarafından gönderilen iş isteklerini nerede bulması, bu istekleri kullanıcı için anlamlı bir arayüz ile nasıl yorumlaması, kullanıcının cevabını iş akış kontrol işlemcisine göndermesi gerektiğini bilmelidir.
- **Yönetimsel Araçlar:** Yeni iş akışları oluşturabilme, varolan iş akışlarında değişiklikler yapabilme, kullanıcıları yönetebilme vb. yeteneklerini sistem yöneticisine sunan araçlardır.
- **İş Akış Veritabanı:** İş akışlarını, bunların adımlarını veya metotlarını, tüm akışları birbirine bağlamak için gerekli kuralları, her adımı hangi kullanıcının gerçekleştireceğini, şu an hangi iş akışının yürütülmekte olduğunu, çalışan iş akışının hangi adımda olduğunu, o adımda kullanıcı tarafından verilen cevabın ne olduğunu bir veritabanında tutmamız gerekmektedir. İş akış veritabanında saklanan bilgiler iki ana başlık altında toplanabilir: iş akış yapısı (kurallar, roller, yönlendirme) bilgileri ve oluşan bilgiler ( kullanıcıdan gelen cevaplar, vb.).
- **Önceden Tanımlanmış Adımlar:** Bir iş akışı adımlardan oluşur. En çok bilinen adımlar şunlardır: başla, kullanıcı cevabı, bildirme, dur.



## 4 ELEKTRONİK TİCARET YAPILARI VE SİSTEMLERİ

İnternetin sağladığı yaratıcılık ve değişiklik ortamında, sanal teşebbüsler veya topluluklar ile sunduğu servisleri birleştirmek isteyen bir ticari kurum, çok çeşitli kurumlar arası elektronik ticaret uygulamalarıyla karşı karşıyadır. Bu bölümde, kurumlar arası elektronik ticaret yapıları, ajan tabanlı elektronik ticaret sistemleri ve antlaşma açıklanmıştır.

### 4.1 Elektronik Ticaret Yapıları

Elektronik ticarete kendi arasında işlenebilirliği sağlamak için bir çok çalışma yapılmaktadır. Bu çalışmaların çoğu XML (Extensible Markup Language) tabanlıdır. Bu konuda en çok ilgi çeken, XML-tabanlı kurumlar arası elektronik ticaret yapılarından bazıları: eCo, RosettaNet ve BizTalk'tur. Bu bölümde bu yapılar, kısaca açıklanmıştır.

#### 4.1.1 eCo

Elektronik ticaret yapılarında ve yazılımlarında birleşmeler pazar tarafından zorunlu olmaktadır. Fakat bu birleşmelerin ne zaman olacağını kestirmek imkansızdır. Bu durumda yeni çalışmaların (specification) geliştirilmesi gerekmektedir. Bu da göstermektedir ki: elektronik ticaret modellerini standart bir yapıda toplamak boş bir çabadır (5). Önemli olan konu, elektronik ticaret yapı ve protokollerinin standartlaşması değil, kurumlar arası uyumluluğun sağlanmasıdır.

Kendi arasında işlenebilirliği katmanlar şeklinde düşünersek, B2B özelliklerini saptamak mümkün olacaktır. Öncelikle, ticari kurumların ne yaptıkları veya nasıl yaptıkları üzerinde ortak bir karara varmak gerekmekte, ne yaptıklarını tanımlayan ortak bir metoda karar verilmelidir. Dahası, elektronik ticaret sistemlerini tanımlayan bu metodun, her sisteme etkisinin tanımlanması gerekmektedir. Potansiyel bir birleşme olması durumunda meta-veri, elektronik ticaret sistem özelliklerini tanımlamak için kullanılabilir. Diğer bir ifade ile bu metod, kendi arasında müzakere edecek iki kurum için, işi nasıl yaptıklarını gösterir. Her ticari kurum için işi nasıl yaptığına ilişkin detaylar perde arkasındadır. Bu da ticari kurumların kendi arasında işlenebilirlik kalitesi açısından farklılaşmayı sağlamaktadır.

Ayrıca dünya üzerinde kendi arasında işlenebilirlik her zaman gerekli değildir. Birçok kurum için beraber iş yapmaya karar verme binlerce karışık karara dayanır. Bir ticari kurumun, bir diğer kurumla iş yapmaya karar vermesi için birçok soruyu cevaplaması gerekir. Kendi arasında işlenebilirlik için temel olan, potansiyel ticari kurum ortağı bulma ve hangi servisleri sunduğunu öğrenmektir. Bundan sonra, bu iş ortağı ile iş yapılıp yapılamayacağını anlamak söz konusu olabilir. Araştırma için bir ticari kurumun sorması gereken sorular aşağıda verilmiştir:

- Hangi ticari kurumları bulabilirim?
- Hangi servisleri sunarlar?
- Ne tür iletişimleri bizden beklerler?
- Hangi protokolleri uygularlar?
- Sistemlerimiz haberleşebilir mi?
- Hangi uygulama arayüzlerini sağlarlar?
- Arayüzlerimiz birbiri ile uyumlu çalışır mı?
- Hangi bilgiler ile mesajlaşmalıyız?

Kendi arasında işlenebilirlik seviyeleri fikirlerine ek olarak ticari kurum servislerine, uygulama arayüzlerini tanımlayan XML dokümanlarını kullanma, eCo düşüncesinin temelini oluşturmaktadır. Bu düşünce, “Doküman Servis Mimarisi” olarak adlandırılmaktadır.

Bu amaçları sağlamak için eCo Framework Projesi Ağustos 1998 yılında, CommerceNet tarafından başlatıldı. Bu projenin asıl sponsoru Commerce One’dır. Bu grubu oluşturan uzmanlar CBL, OBI, OFX, IOTP, XML/EDI, RosettaNet, URN, HTML, W3C XML çalışmalarının özelliklerini belirlemişlerdir. Bu yapı, tüm elektronik ticaret (EDI, OBI ve XML-tabanlı dikey market kelimelerini de içermek üzere) için ortak bir yapıyı araştırmaktadır.

Bu yapı 4 katman üzerine kurulmuştur:

- Özel internet servisleri (dikey marketler),
- Ticari servisler (elektronik ticaretin genel tipleri),
- Ticaret servisleri (ödeme mekanizmaları ve güvenlik protokolleri),
- Ağ servisleri (altyapı).

#### 4.1.1.1 Tasarım ilkeleri

Elektronik ticaretin kendi arasında işlenebilirliğini sağlamak amacını elde etmek için aşağıdaki temel tasarım ilkelerine dikkat edilmelidir:

- İnternet standartlarını ve özelliklerini kullanmak,
- Varolan veya yeni oluşturulacak standartları kullanmak,
- Kendi arasında işlenebilirliği ilerletmek,
- Uluslararası dil desteğini sağlamak,
- Genişleyebilirlik,
- Basitlik,
- Uygulanabilirlik.

#### 4.1.1.2 Gerekenler

Heterojen elektronik ticaret sistemlerinin kendi aralarında işlem yapabilmeleri için gerekenler aşağıda verilmiştir:

- Elektronik ticaret sistemlerinin tanımlanması için ortak bir temel yapı gerekmektedir.
- Kurulacak yapı, internet üzerinden elektronik ticaret sistemlerini bulabilmelidir.
- Elektronik ticaret sistemlerinin kullandığı haberleşme protokollerini tanımlayan ortak bir metot sağlanmalıdır.
- Yapı, elektronik ticaret tanımlamalarını, dil özelliklerini tanımlayan bir mekanizma sunmalıdır.
- Yapı, bir ticari kurumun sunduğu teklifleri diğerleri ile birleştirebilmelidir.
- Yapı, iş süreçleri ile organizasyonlar arasında köprü kurabilmelidir.
- Bu yapı ile tasarlanmamış yapılar için, bu yapı ile iletişimi sağlayan arayüzler olmalıdır.

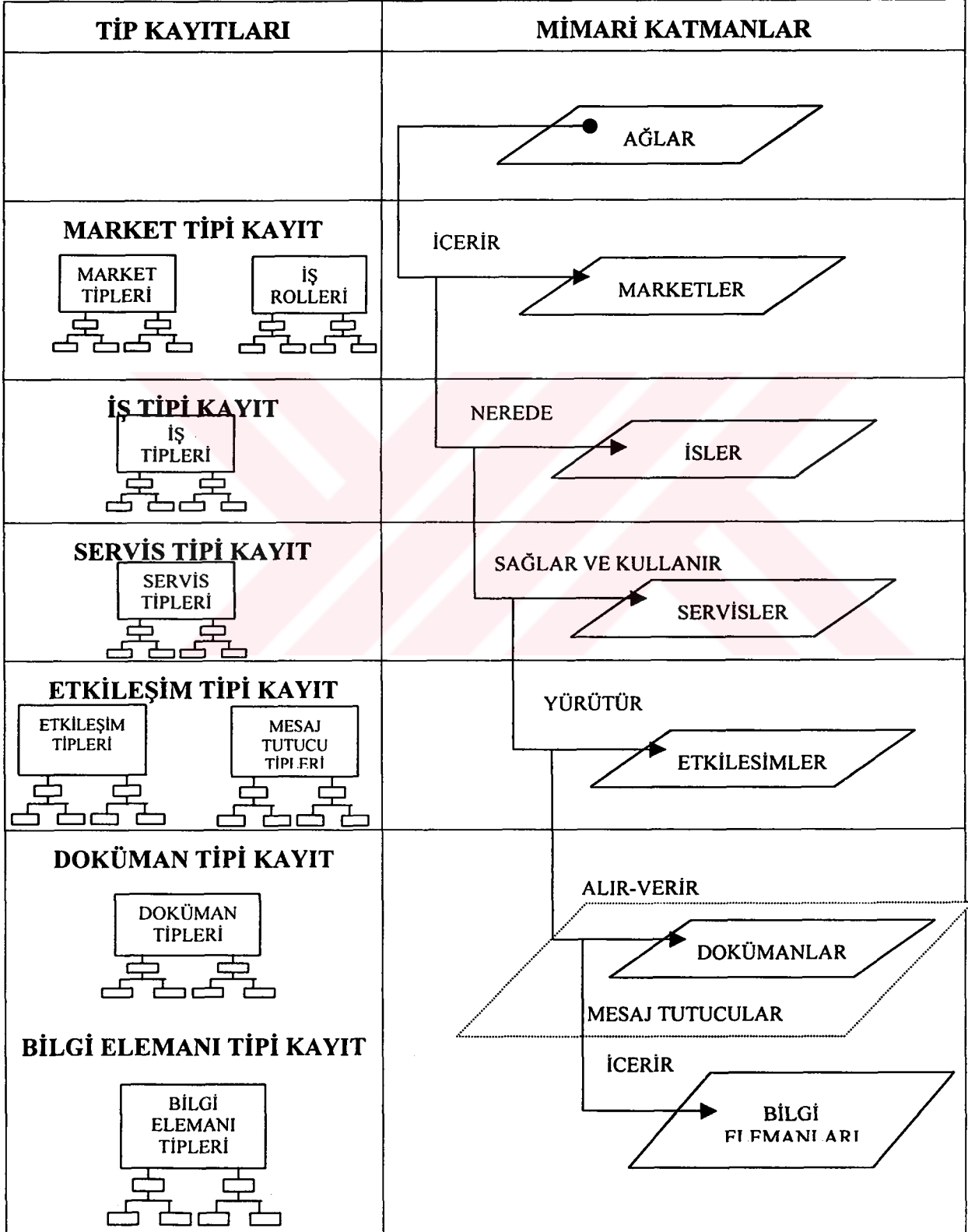
#### 4.1.1.3 eCo yapısı

B2B ticaretin olması için internet üzerindeki elektronik ticaret sistemleri arasında olabilmesi için ticari ortaklar, aşağıdaki temelleri sunabilmelidir:

- İnternet üzerindeki diğer ticari kurumları bulabilmelidir.
- Ortak iş yapmak isteyip istemediklerini ve pazara nasıl katılacaklarını anlayabilmelidir.
- Diğer ticari ortağın sağladığı ve istediği servisleri belirleyebilmelidir.
- İki kurum arasındaki etkileşimleri ve bu etkileşimlerde hangi dokümanları ve bilgileri kullandığını belirlemelidir.
- Elektronik ticaret sistemlerinin nasıl haberleşeceklerini belirlemelidir.
- Sistemler arasında işlenebilirliği sağlamak için ne gibi değişiklik yapılacağını belirleyebilmelidir.
- Eğer istenirse haberleşme, internet yerine kanallar vasıtası ile yapılmalıdır.

#### 4.1.1.4 eCo yapı katmanları

eCo yapısının her katmanındaki tip kayıtları ve katmanlar arasındaki ilişki, Şekil 4.1'de gösterilmiştir.



Şekil 4.1 eCo yapısının katmanları

eCo yapısı teknik ilkeler yerine, iş ilkelerine göre kurulmuştur. Ticari kurumlar kavramsal eCo yapısı ile elektronik ticaret sistemlerinin meta-verilerini tanımlayabilirler ve yayınlatabilirler. Meta-veri sayesinde elektronik ticaret sistemlerinin birçok etkisini ticari ortaklarının anlamasını ve kendi arasında işlenebilirliği sağlarlar.

eCo yapısı, elektronik ticareti katmanlı olarak modeller. Her katman bir sonraki katmanla tanımlanmış şekilde etkileşim kurar. Bilgi veya özellikler yani meta-veri, bu yapı ile her katman için ayrı tip kayıtçıları ile birbirinden farklıca tanımlanabilir. Ticari ortaklık kurmak isteyen kurumlar için, her katmandaki bilgi ile kurumun sunduğu servisler hakkında bilgi elde edilir veya pazara girilir veya pazar yerindeki diğer kurumlarla ortak çalışılır veya yeni bir servis sunulur.

eCo yapısında en üst katman, “Ağlar”dır. Ağlar, elektronik ticaretin olduğu bir veya birden fazla ağı ifade eder. Bu ağlar, birçok pazar yerini veya pazarı özel malları veya servisleri sağlamak için veya elde etmek için içerebilir.

Ağıdaki her market bağımsız olabilir veya markete katılmak için kendi kuralları, prosedürleri ve protokolleri olabilir. Marketler işlerden oluşur. eCo yapısında bu, “işler” katmanı olarak tanımlanmıştır. Örneğin bu katman için tipini, yerini, web sayfasını ve diğer ilgili bilgileri belirleyebilir.

Herhangi bir iş birden fazla markete katılabilir. Örneğin bir bilgisayar üreticisi donanım, yazılım, finans, danışma, servis marketlerine katılabilir. Diğer bir deyişle, iş içindeki her aktivite veya fonksiyon o aktivitelerin gereklerini sağlamak amacıyla markete katılabilir. Ayrıca, bu marketler sadece bir işe özel olabilir veya birçok iş ile beraber çalışabilir. Örneğin bir iş, üçüncü parti işin bulunduğu endüstride dikey olarak üçüncü parti tedarik zincirlerine katılabilir.

eCo yapısının bir sonraki katmanı “servisler” katmanıdır. Bu her işin sunduğu servisin tiplerini, arayüzlerini ve bu servisi sunmak için gerekli bilgileri sağlar. Katalog tarama, ürün sipariş verme, ödeme yapma, sipariş durumunu kontrol etme, ürün yaşam zinciri, yeni ürün tanıtımı gibi işlemleri içerir. Her iş servisi için arayüzler, servisler katmanında tanımlanır.

İş sağlayanlar arasında servis tipleri ve bunların arayüzleri farklı şekilde tanımlanabilir. Fakat, pazardaki bir grup ortak iş ortak bir iletişim kurmak için tanımlama yaparlar. Örneğin dikey bir endüstride bu endüstriye katılanlar ortak katalog güncelleme servisi, ürün verisi değişimi, hatta analiz servisi gibi servislerde ortak tanımlamalar yaparlar.

Her sağlanan servis, tüm servisin birçok opsiyonunu sağlayan alt servislerden oluşur. Bu yapıda alt servisler de “servisler” katmanında tanımlanır.

“Etkileşim” katmanı, servis kullanıcıları tarafından görülmez. Bir servis çağrıldığında, gerekli etkileşimin koreografisini tanımlar. Bu koreografide olayların gerçek sırası önceden belirlenebilir veya servis tarafından sunulan, kullanıcının özel opsiyonlar arasındaki seçimi sayesinde olay-yürütümlü olabilir. Bu katman her servisin arkasındaki etkileşimleri, her etkileşim sırasında hangi tip mesajların değiş-tokuş edildiğini tanımlar. Her mesaj veya mesaj tutucu her etkileşim sırasında gereken birden fazla doküman tipini veya bilgiyi içerir. Etkileşim sırasında değiştirilen doküman tipleri, eCom yapısında “Dokümanlar” katmanı olarak tanımlanır. Son katman, “Bilgi Elemanları” bilgi elemanı tiplerini yani veri elemanları ve özelliklerini tanımlar.

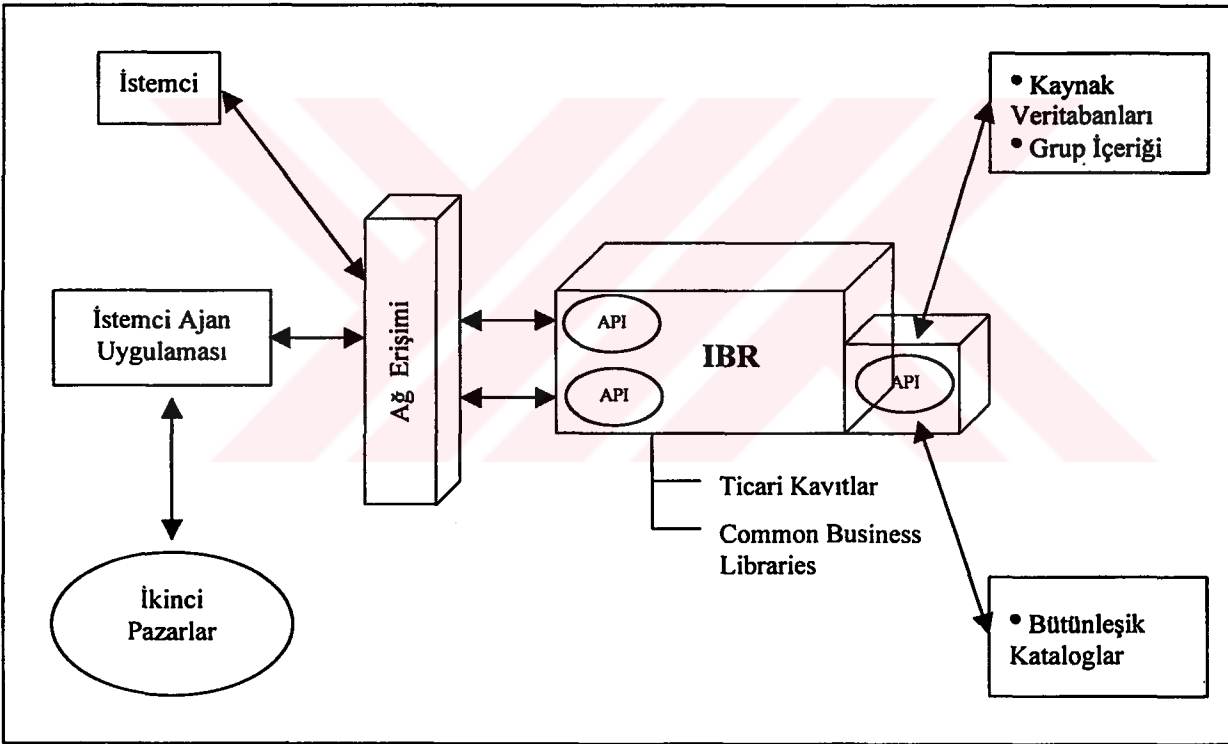
Bir iş kendi veri elemanlarını ve dokümanlarını belirleyebilir. Bir iş üçüncü parti, özel bir domain veya standart tarafından tanımlanan veri elemanlarını ve dokümanlarını kullanabilir.

eCo yapısı ayrıca her katmandaki bilgiyi veya özelliği sorgulamak için arayüzler tanımlar. Her katmanın yayınlanan bilgilerini ve özellik kümesini kontrol ederek ilgilenen kurumların, o katman hakkında bilgi elde etmesi sağlanabilir.

eCo pazarlarında birden fazla satıcı veya ürün veya servis üreticileri ve diğer tarafta da birçok alıcı bu ürünleri birbiri ile karşılaştırıp en iyi şekilde alım kararı vermek isteyeceklerdir. Bu üreticilerin ürünleri birçok sunucu üzerinde ve farklı veritabanlarında olacaktır. Veritabanları birbirleri ile açık güvenilir kayıtlamayla birleştirilecektir. Kayıt şemaları ve veri elemanları, XML tabanlı Common Business Library (CBL)’e kaydedilecektir. Bu kayıtlar, satıcıların ürünlerini/servislerini kaydetmeyi, kendilerinin ve servislerinin doğruluklarını kanıtlamayı, diğer satıcılarla ürünlerinin karşılaştırılmasını sağlayacaktır. Alıcılar kendi seçimlerini, profillerini, yetkilerini özel bir şekilde yapabileceklerdir. Bu bütünleşme ve kaydetme

yetenekleri ticari ürünler, servisler ve süreçler için çevirme ve anlam eşleme tabanlı sunucu araçlarına dayalı olarak gerçekleştirilecektir.

eCo pazarı ikinci pazar yaratabilen ve pazardaki katılımcıların isteklerine göre etkileşimde bulunan CBL tabanlı kayıtları kullanabilen ajan yapısına da sahip olacaktır. XML'in web ajanlarını ve diğer otomatik süreçlere olanak sağlaması gibi, ticaretin doğasını da çevirmelidir. Tüketiciler için akıllı alışveriş, ajanların olması ile sağlanacaktır. Intelligent Business Registries (IBR) bu ajanların istenen ürünleri, servisleri veya süreçleri bulmasını sağlayacaktır. Ajanlar bu kaynakları paralel olarak sorgulayabilecek, böylece en iyi fiyatı veren veya ilişkiyi bulacaktır (Dills, 1999). Şekil 4.2'de IBR bileşenleri verilmiştir.



Şekil 4.2 IBR bileşenleri

#### 4.1.1.4.1 CBL

Uygulama ajanlarının birbirleri ile konuşabilmesi için CBL geliştirilmektedir. CBL, en azından şu ticari kavramlar için bilgi modelleri içermelidir (1):



- Ticari temel tanımlar: kurumlar, servisler ve ürünler,
- Ticari formlar: kataloglar, alım emirleri ve faturalar,
- Standart ölçüler: zaman ve tarih, yer,
- Sınıflama kodları.

#### 4.1.1.5 eCo katmanları ve kayıtları

eCo yapısı birçok satıcının ve alıcının en fazla bilgi ile karar verebileceği özellikleri içermektedir. Kayıt şemaları, XML-tabanlı veri elemanları ile oluşturulmuştur. Satıcılar ürünlerini/hizmetlerini kaydedebilir, kendilerinin ve hizmet politikalarının gerçekliklerini kanıtlayabilirler. Alıcılar tam olarak ürünleri karşılaştırabilirler. Satıcı ve alıcılar isteklerini, profillerini halka açık ve özel olarak kaydedebilirler.

Bir eCo Document Wrapper'ının eCo-XML dokümanını tanıyabilmesi için aşağıdaki bilgileri içermelidir.

Head(zorunlu)  
 Identifier(zorunlu)- URI içerir.  
 Creator(seçmeli)  
 Date(seçmeli)  
 Version(seçmeli)  
 TimeToLive(seçmeli)  
 Description(seçmeli)  
 Label(seçmeli)

Şekil 4.3'te örnek bir eCo dokümanı verilmiştir.

```
<?xml version="1.0"?>
<myDocumentType xmlns='http://www.commerce.net/eco'>
<Head>
  <Identifier>http://www.commerce.net/...</identifier>
  <Creator>http://www.commerce.net/~bob</Creator>
  <Date>19990628</ Date>
  <Version>1.0</ Version>
  <TimeToLive>86400</ TimeToLive>
  <Description>Açıklama.</ Description>
  <Label>Doküman</ Label>
</Head>
...
</myDocumentType>
```

Şekil 4.3 eCo doküman örneği

Şu ortak sorgular her katman tarafından sağlanmalıdır. GetSupportedVersions(zorunlu), QueryInterface(zorunlu) dönen dokümanlar için: EcoVersionList, EcoInterfaceList...

Ağ katmanında tanımlanması gereken sorgular ise NetworkGetProperties(zorunlu), NetworkGetMarkets(seçmeli),...Cevap olarak: NetworkPropertySheet...

Bir kurumun eCo uyumlu olabilmesi için web sitesinin kök elemanını aşağıdaki XML ile değiştirmesi gerekmektedir.

```
<?xml version="1.0"?>
<EcoInterfaces xmlns="http://www.commerce.net/eco">
  <Head>
    <identifier>http://www.commerce.net/...</identifier>
    <creator> http://www.commerce.net/~peri</ creator><Date>20010603</ Date>
    <version>1.0</ version>
    <TimeToLive>86400</ TimeToLive>
    <Description>...</ Description>
    <Label>...</ Label>
  </Head>
  <Interface type="Business">
    <identifier>http://www.mycorp.com/...</identifier>
    <creator> http://www.whoever.com</ creator>
    <version>1.0</ version>
    <Date>20010603</ Date>
    <TimeToLive>86400</ TimeToLive>
    <Description>Business interface tanımları</ Description>
    <Label>İşim</ Label>
  </Interface>
  <Interface type="Service">
    <identifier>http://www.comerce.net/eco/OrderService</identifier>
    <creator> http://www.commerce.net/~peri</ creator>
    <version>1.0</ version>
    <Date>20010603</ Date>
    <TimeToLive>86400</ TimeToLive>
    <Description>Talep Tanımlarım</ Description>
    <Label>İşimin talep tanımları</ Label>
  </Interface>
  <Interface type="Service">
    <identifier>http://www.comerce.net/eco/CatalogService</identifier>
    <creator> http://www.commerce.net/~peri</ creator>...
  </Interface>
</EcoInterfaces>
```

Şekil 4.4 eCo uyumlu dokümanın kök elemanı

#### 4.1.2 RosettaNet

Elektronik ticaret tanımlarını geliştiren katmanlı şema tutucudur. RosettaNet, ticareti destekleyen üç katmandan oluşmaktadır:

- Sözlükler,
- Yapılar (frameworks),
- Partner Interface Processes (PIP).

Bir sözlük, iş kesilemez uygulamalarını tanımlayan özellikler veya kelime gruplarıdır. RosettaNet'in iki veri sözlüğü bulunmaktadır. Birincisi, ürünleri tanımlamaktadır; ikincisi ise, katalogları, ortakları ve kesilemez uygulamaları tanımlamaktadır. Daha sonra bu kelimeler ve özellikler bir gramer oluşturmak için değişim protokolleri ile birleştirilecektir.

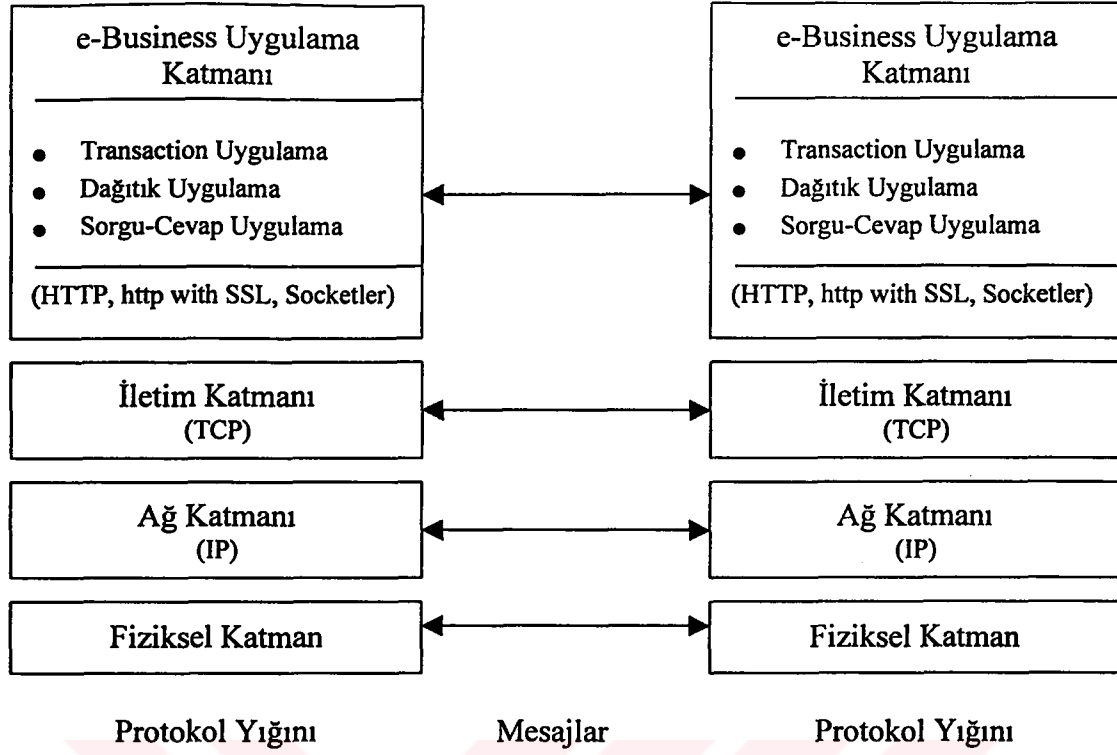
RosettaNet'in en çok önem verdiği konu, PIP kavramıdır. PIP'yi, iki iş ortağı arasındaki diyalog olarak tanımlamaktadır. Bir PIP, bir dizi XML değişim dokümanları ve iş ortaklarının nasıl iletişim kurduğunu tanımlayan bir iş modelinden oluşmaktadır. Ayrıca PIP dokümanların geçerliliğini kontrol eden bir mekanizma da sunmaktadır.

Her PIP, farklı ticari organizasyonlarda çalışan iki sürecin nasıl standartlaştırılacağı ve tedarik zincirlerinde arayüzlerinin ortaya koyulmasını belirler. PIP, tüm iş mantığını, mesaj akışını ve mesaj içeriğinin iki süreç arasında işbirliği yapmasını sağlar. PIP, 17 segment 6 cluster'dan oluşur. Cluster ve segmentler tedarik zinciri süreçlerini birleştirerek daha yönetilebilir yapı sağlar. Cluster ve segmentler, Şekil 4.5'te verilmiştir.

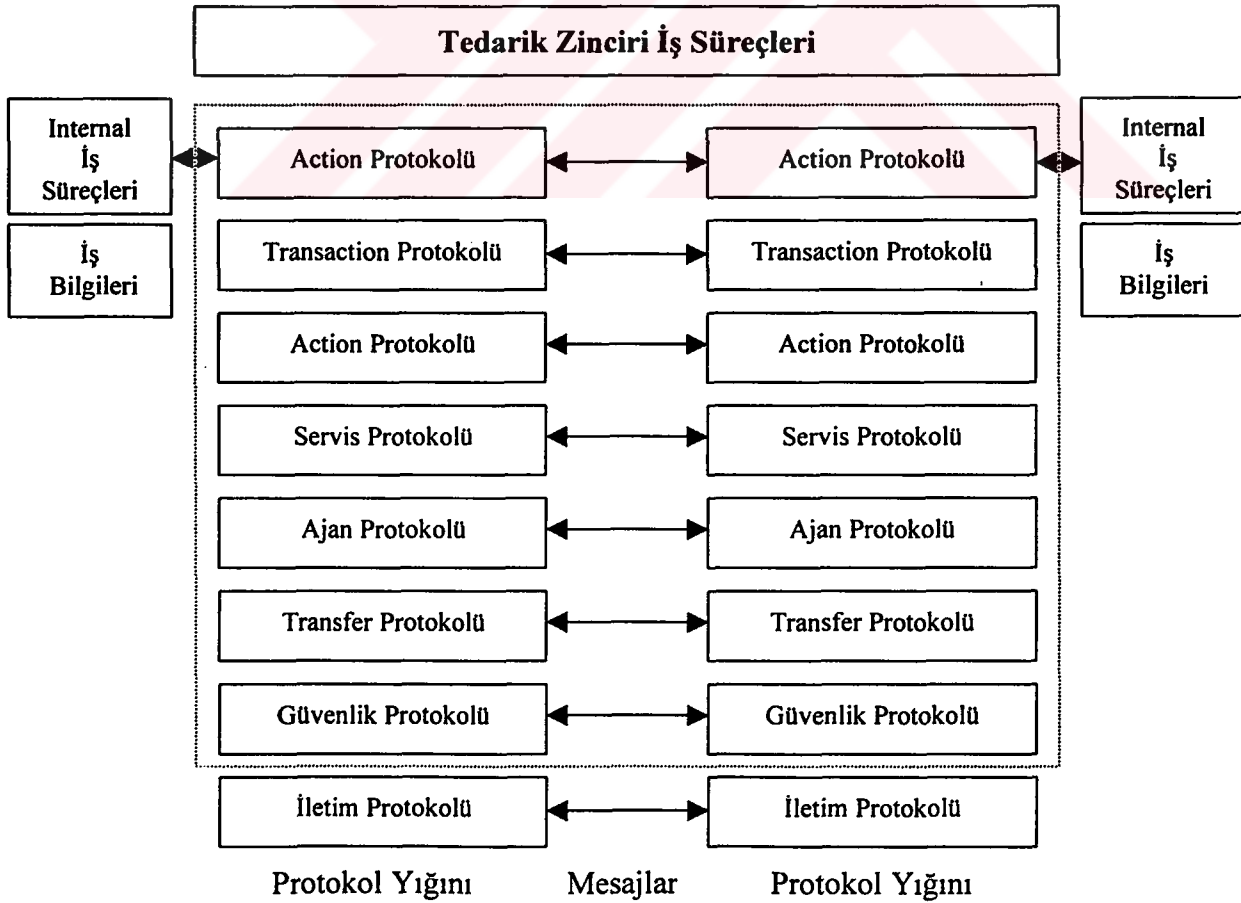
- Cluster 1 : Partner, ürün ve hizmet
- Segment 1A Partner review
  - Segment 1B Ürün ve hizmet review
- Cluster 2 : Ürün tanıtımı
- Segment 2A Dağıtım için hazırlanma
  - Segment 2B Ürün değişimi bildirim
- Cluster 3 : Sipariş yönetimi
- Segment 3A Piyasa fiyatını bildirme (quote) ve sipariş girişi
  - Segment 3B Ulaşım ve dağıtım
  - Segment 3C Returns ve finans
  - Segment 3D Ürün konfigürasyonu
- Cluster 4 : Mal sayımı yönetimi
- Segment 4A Ortak tahmin
  - Segment 4B Mal tahsisi ve ikmal
  - Segment 4C Mal sayımı raporlaması
  - Segment 4D Mal ikmal
  - Segment 4E Satış raporlaması
  - Segment 4F Fiyat korunumu
- Cluster 5 : Pazarlama bilgisi yönetimi
- Segment 5A Sevketme/avantaj yönetimi
  - Segment 5B Pazar kampanya yönetimi
  - Segment 5C Kazanma yönetiminin tasarımı (elektronik parçalar için)
  - Segment 5D Stoktan ve zimmetten gönderme (elektronik parçalar için)
- Cluster 6 : Hizmet ve destek
- Segment 6A Garantileri, servis paketlerini, sözleşme servislerini sağlama ve yönetme,
  - Segment 6B Fayda yönetimini sağlama ve yönetme
  - Segment 6C Teknik destek ve hizmet yönetimi

Şekil 4.5 RosettaNet cluster ve segment adları

RosettaNet elektronik iş uygulama yapısı Şekil 4.6'da açıklanmıştır. RosettaNet bilgi teknolojisi (IT), elektronik parçalar (EC) ve yarıiletken üretimi (SM) tedarik zincirleri için standartlar geliştirmektedir. Şekil 4.7'de, RosettaNet'in tedarik zincirleri süreçlerini de kapsayan protokol yapısı gösterilmiştir.



Şekil 4.6 RosettaNet elektronik iş uygulama yapısı



Şekil 4.7 Protokol yapısı

### 4.1.3 BizTalk

Kurumlar arası ticaret gerçekleştirilirken kurumlar, kendi içindeki standartlara göre birbirlerine mesaj göndermektedir. Bu mesajların her biri, birçok gerekli gereksiz veri içerdiğinden bunların anlamlı bir şekilde birbirine çevriminin gerçekleştirilmesi gerekmektedir. Bu standartlaşmayı EDI gerçekleştirmiştir fakat, bu standartlar çok karmaşıktır ve kurumların bu standartları gerçekleştirmesi çok pahalıya mal olmaktadır. Ayrıca kurumlar mesajlaşırken, farklı haberleşme protokolleri kullandıklarından dolayı uyumluluk problemi ile karşı karşıya kalmaktadır. Tüm bu nedenlerden dolayı, yakın zamana kadar uygulama entegrasyonunu çok büyük kurumlar gerçekleştirmiştir.

Bu durumda gerekli olan, standart endüstri mesaj şemaları geliştirmeyi sağlayan bir yapıdır. Daha sonra BizTalk Server, mesajlaşma sistemini ayarlamak için bu şemaları temel olarak kullanacaktır. Bu yapıyı kullanmakla; oldukça basit fakat şema geliştirilmesi sırasında, kurumların ortak iş yapmasına esneklik sağlayan uyumlu mesaj şekillerini geliştirilmesi anlaşılmalıdır. Ayrıca bu yapının gerçekleştirimi, küçük ve orta boy kurumların uygulama entegrasyonuna katılması için oldukça ucuz olmalıdır.

Yukarıda anlatılanlar BizTalk Framework Initiative'in amaçlarıdır. Bu yapı, Microsoft tarafından geliştirilmektedir ve herkese açıktır. Herhangi bir işletim sistemi veya yazılım ön şart değildir. BizTalk Framework Initiative, üç parçadan oluşmaktadır:

- **BizTalk Yapısı:** Organizasyonlar ve uygulamalar arasında güvenli mesaj değişimi için bir dizi kurallardır.
- **BizTalk.org:** Mesaj değişimi ve ilgili teknolojiler ve mesaj şema tutucuları hakkında bilgi sağlayan web sitesidir.
- **BizTalk Server:** BizTalk uyumlu mesaj dokümanlarını işleyen yazılımdır.

#### 4.1.3.1 BizTalk yapısı

BizTalk yapısı, platform ve teknolojiye bağımsız olarak kurumlar arasında uyumluluğu sağlamaktadır. Uygulamalar ve organizasyonlar arasında haberleşme için; XML-tabanlı mesajlaşma çözümlerinin tasarımını ve geliştirilmesini sağlar. Sağladığı bu özellikler standart

ve yeni çıkan internet teknolojileri, örneğin Hypertext Transfer Protocol (HTTP), Multipurpose Internet Mail Extensions (MIME), Extensible Markup Language (XML) ve Simple Object Access Protocol (SOAP) üzerine kurulmuştur.

BizTalk yapısı kurumlar arası elektronik ticaretin tüm yanlarını içermemektedir. Örneğin, doğrudan hukuki, hakem usulü anlaşmaları, sistemin çökmesi durumunda bilgileri kurtarmayı, alış ve ticarete güvenlik için gerekli olan özel iş süreçlerini tanımlamaya izin vermemektedir. BizTalk yapısı, kurumlar arası elektronik ticaret için temel olan mekanizmaları sağlamaktadır; doküman değişimi için mesaj şemaları yaratma kurallarından oluşmaktadır. BizTalk yapısı iki açık standarda dayanmaktadır:

- Extensible Markup Language (XML)
- Simple Object Access Protocol (SOAP)

#### 4.1.3.1.1 XML

XML gerçek bir dil değildir fakat, işaretleme dillerini yaratmamızı sağlayan bir standart sözdizimidir. XML açık, okunaklı isteklerinize uygun bir işaretleme dilidir. Bu yüzden XML-tabanlı yapı; uyumlu, tanımlayıcı ve açık olmasından dolayı mesajlar için idealdir.

#### 4.1.3.1.2 SOAP

Platform-özel veri erişim problemlerini çözmek için Simple Object Access Protocol teklif edilmiştir. SOAP, veriyi koyduğunuz elektronik zarfı tanımlayan XML-tabanlı sözdizimidir. Uzaktan metotları çağırmanızı sağlayan uygulamaları gerçekleştirmemize olanak tanımaktadır. Bir SOAP paketi, aşağıdaki şekilde kullanılan SOAP istemci ve sunucudan oluşmaktadır:

- SOAP istemci, SOAP istek dokümanını yaratır ve HTTP üzerinden diğer sitede bulunan sunucuya verir.
- SOAP sunucu mesajı alır, SOAP dokümanını çözümler, kendi domainindeki uzak nesnedeki metodu çağırır.

- Uzak nesne; bilgiyi, SOAP dokümanına koyan ve SOAP istemcisine gönderen sunucuya gönderir.

Bu yüzden, SOAP sunucu yorumlayıcı (interpreter) olarak davranır ve SOAP paketi iki uyumsuz protokol arasında arayüz sağlar.

#### 4.1.3.1.3 BizTalk mesajları

Bir BizTalk dokümanı, BizTalk-özel XML etiketlerini içeren bir SOAP dokümanı uzantısıdır. BizTalk yapısını gerçekleştiren sunucunun, yapıda tanımlanan BizTag'ları tanıması ve buna göre gerekeni yapması beklenmektedir. BizTalk yapısı geniş olarak SOAP standardını temel almıştır ve bu yüzden de BizTalk mesajları, XML ve SOAP'ın avantajlarını sağlamaktadır: genişleyebilirlik, açıklık ve platform-bağımsızlık. BizTalk'ta gerçekleştirilen mesajlaşmanın daha iyi kavranabilmesi için, mesajlaşmada kullanılan elemanların tanımının verilmesi gerekmektedir.

#### 4.1.3.1.3.1 Terminoloji

**BizTalk Yapısına Uyumlu Sunucu (BFC):** BFC (BizTalk Framework Compliant) sunucusu, BizTalk yapısında tanımlanan mesaj işleme fonksiyonelliğini sağlayan servis kümesi olarak tanımlanır.

**Uygulama:** İş verilerinin veya mantığının saklandığı ve yürütüldüğü iş sistemidir. Ayrıca bir uygulama, iş dokümanlarını yaymak veya tüketmek, BFC sunucusu ile haberleşmek için gerekli adaptörleri de içerir.

**İş Dokümanı:** İş verisini tutmak için iyi oluşturulmuş XML dokümanıdır.

**Şema:** XML doküman sınıfının yapısını tanımlayan meta veridir. Şemalar BizTalk etiketlerinden oluşmamakta, sadece işin yapılması için ticari kurumların ortak olarak tanımladıkları bilgi etiketlerinden oluşmaktadır.



**BizTalk Dokümanı:** SOAP 1.1 mesajdır. Mesajın gövdesi iş dokümanlarını, mesajın başlığı gelişmiş mesaj-çözme anlam kümeleri için BizTalk'a özel başlıklar içerir. Aşağıdaki kavramlar BizTalk dokümanları tarafından kullanılır:

**Lifetime:** Dokümanın geçerli olduğu süreyi belirler.

**Identity:** Dokümanı belirlemek için kullanılan numara.

**Acceptance:** Alıcı tarafından kabul edilme işlemidir.

**Idempotence:** Bir dokümanın birden fazla gönderilmesi ve kabul edilmesi ile sadece bir defa gönderilip, alınması durumunda aynı tepkiyi verme yeteneğidir.

**Receipts:** BizTalk yapısı uç-uca protokolleri içerir. Sadece iki tipi vardır:

*Delivery:* Alıcının dağıtılan dokümanı aldığını belirtir.

*Commitment:* Sadece dokümanın alındığını değil, ayrıca tüm mesaj içeriğinin, mesajın doğru olup olmadığının kontrol edildiğini belirtir.

**BizTag:** BizTalk dokümanlarını çözmeye zorunlu veya opsiyonel XML etiketleridir.

**BizTalk Mesajı:** BizTalk mesajları BFC sunucular arasında BizTalk dokümanlarını, ilgili dosyaları göndermek için kullanılır. BizTalk mesajı, mesajın anlamını belirleyen BizTalk dokümanını içermelidir.

**Transport:** Bir uçtan bir uca bilginin iletimini ifade eder.

**Endpoint:** BizTalk yapısı ile uyumlu, BizTalk mesajının kaynağı veya hedefi.

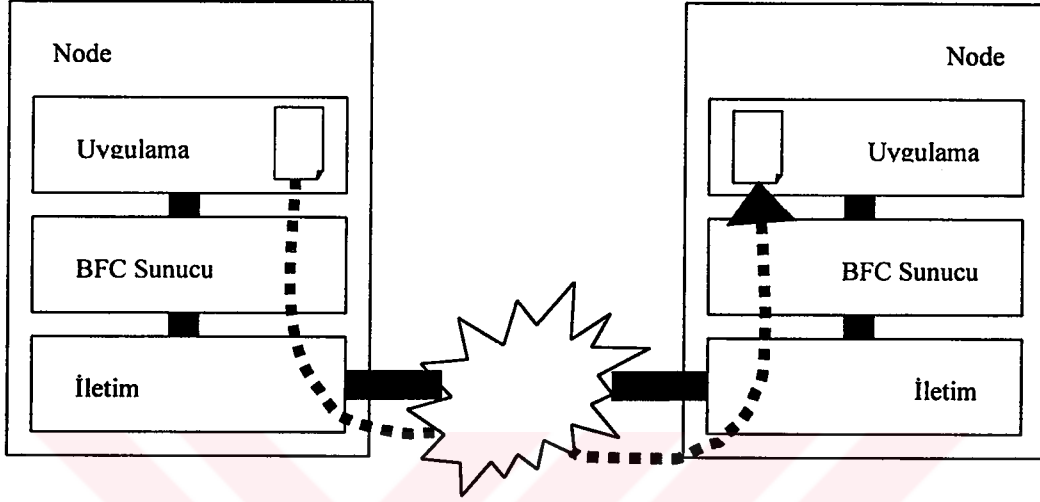
**Address:** Endpointin yerini ifade eder.

**Attachment:** BizTalk dokümanının gövdesi içinde gönderilmeyen, XML-olmayan eklerdir.

#### 4.1.3.1.4 Mantıksal katmanlar

BizTalk yapısının mantıksal katmanları üç tanedir. Buradaki katmanlama sadece gösterim ve açıklama amaçları için kullanılmıştır. Bu katmanlama Şekil 4.8'de verilmiştir. Birden fazla BFC sunucusu birçok haberleşme protokolü (HTTP, SMTP, Microsoft Message Queue[MSMQ]) kullanır.

Bir uygulama diğer bir node'a göndermek üzere iş dokümanlarını ve eklerini yaratmadan, BFC sunucuya vermeden sorumludur. İş dokümanlarını açmadan BFC sunucusu yapısına göre, uygulama veya BFC sunucusu sorumludur. Sunucu dokümanı ve eklerini işler ve transport katmanı için BizTalk mesajını oluşturur (Microsoft BizTalk Server, 2000).



Şekil 4.8 Mantıksal katmanlar

BizTalk'da sistemler, uygulamalar arasında mesajlaşma üzerine kurulmuştur. Böylece her adımda hangi mesajın gönderileceğini ve alınacağını belirlenir. İş akışları, verinin şekil değiştirmesi ve içeriğinin değiştirilmesi olarak algılanır. Bu durumda, verinin akışı uygulama olmaktadır. Veri akışı da BizTalk tarafından sağlanmaktadır.

İş akışlarının tasarımı ve gerçekleştirimi orchestration olarak bilinir. Orchestration Tasarımcısı, Microsoft Message Queue ile mesaj alışverişini sağlar. Transaction'ları destekler ve iki veya daha fazla sürecin aynı anda çalışmasını sağlar. İş akışları, XLANG dilinde kaydedilmektedir.

#### 4.1.3.2 BizTalk.org

BizTalk yapısı, dokümanlarının kullanımını ve genel olarak XML-tabanlı mesajlaşmayı yaymak için, BizTalk.org web sitesi ([www.biztalk.org](http://www.biztalk.org)) kurulmuştur. Bu site, Microsoft tarafından yönetilmektedir ve BizTalk yapısı, BizTalk grubu hakkında haberler, tartışmalar ve uygulama entegrasyonunu sağlayan araçlar hakkında bilgi içermektedir.

Ayrıca bu site, BizTalk yapısını gerçekleştiren kurumların mesaj şekillerini tutan XML şemalarını da içermektedir. Şemasını yayınlamak isteyen herkese ve yayınlanmış bir şema bulmak isteyen herkese açıktır. Ayrıca bu site sayesinde, iyi-tasarlanmış değişim şekilleri hakkında programcılar eğitilmektedir.

Uzun vadede ise bu şema saklayıcı, B2B birleştirici portal'inin temel taşı olacaktır. Potansiyel iş ortakları satıcıların şemalarını buradan elde edecekler ve entegrasyon çalışmalarını başlatacaklardır.

#### **4.1.3.2.1 BizTalk şema tutucu**

Başarılı bir kurumlar arası elektronik ticareti gerçekleştirmek için, olası iş ortağı önce uygun bir iş ortağını, daha sonra da ne tür şemaları kullandığını bulmalıdır. İş ortağı, ya bu şemaları olduğu gibi kullanacak ya da kendi şemaları ve yeni iş ortağının şemaları arasında bir eşleme yapacaktır. BizTalk Server bize bu en son basamak için araçlar sunmaktadır. Merkezi elektronik ticaret şemaları ilk iki adımı tanımlamaktadır.

BizTalk.org web sitesi böyle bir kütüphane sunmaktadır. Olası iş ortağı bu siteden endüstri kategorisine göre şemaları arayabilmektedir. Daha sonrası North American Industry Classification System (NAICS)'e göre tanımlanmıştır. Bu sistem, işi, işin ait olduğu endüstri grubuna veya kategorisine göre sınıflayıcı kodlar üzerine kurulmuştur. Bu iki sistem kullanılarak olası müşteri, az sayıda şemayı belirlemekte ve isteklerine uygun olup olmadığını araştırabilmektedir.

Uygun şemalar bulunduktan sonra iş ortağı bunları, download etmekte ve detaylı bilgiyi incelemektedir. Sitede bulunan her şemanın üç dokümanı bulunmaktadır: şema, HTML-tabanlı programlama dokümanı ve şemayla uyumlu bir mesaj.

## 4.2 Elektronik Ticaret Sistemleri

Üçüncü bölümde açıklanan teknolojiler kullanılarak aşağıdaki elektronik ticaret sistemleri uygulamaları gerçekleştirilmiştir (Yılmaz, 1999).

### 4.2.1 Ajan aracılı elektronik ticaret

MIT(Massachusetts Institute of Technology) görüşüne göre akıllı ajanlar kendi kendini yöneten araçlar, elektronik ticarete önemli rol alacaklardır. Akıllı ajanlar bir görevi yürüten programlardır. Elektronik ticaretin güvenlik, kanun, ödeme mekanizmaları, reklam verme, ortak kavramlar, online kataloglar, arbulucular gibi çok çeşitli konuları içermesi, ajan teknolojilerinin sürekli olarak çalışan, kişiselleşmiş, yarı kendi kendini yöneten davranış istendiğinde kullanılmasını sağlamıştır. MIT, yaptıkları projede, çok çeşitli arbulucuların olacağını belirtmekte ve elektronik ticareti B2B, B2C ve C2C olmak üzere üç şekilde ifade etmektedir. MIT'ye göre tüketici alım davranışları altı basamakta belirtilebilir:

- İhtiyaç belirlenmesi; tüketicilere ürün bilgilerinin verilmesi problemin algılanması,
- Ürün komisyoncuları; hangi ürünün alınacağına yardım eden bilginin elde edilmesi,
- Tüccar aracılığı; tüketicinin seçtiği kriterlere göre hangi satıcıdan, hangi ürünü alacağını belirlenmesi,
- Anlaşma; işlemin terimlerinin belirlenmesi,
- Alım ve dağıtım,
- Servis hizmetleri.

MIT, Kasbah ve Tete-a-Tete olmak üzere iki sistem geliştirmiştir.

#### 4.2.1.1 Kasbah

C2C elektronik ticareti gerçekleştirir. İki yönlü artırım mekanizmasına dayanır. Kasbah'da satıcı ajanı, bir reklam olarak düşünürler. Tüketici, alıcı ajanı yarattığı zaman, hangi malı satacağına dair tanımı verir. Geleneksel sınıflanmış reklamlardan farklı olarak, proaktif yapıdaki reklamlardır. İlgili taraflarla iletişime geçerek en iyi fiyat için antlaşmaya çalışırlar.

Tüketici, yeni bir satıcı ajan yarattığında istediği malı almak için, amacını belirten birkaç parametre tanımlar. Bu parametreler; malın satılacağı tarih, istenilen fiyat ve en düşük kabul edilebilir fiyattır.

#### 4.2.1.2 Tete-a-tete

Bu çok ajanlı bir sistemdir ve yukarıda bahsedilen tüketici alım davranışlarından sadece üçünü kapsar. Bu sistemde karşılaştırma sadece fiyata göre yapılmayıp; dağıtım zamanı, destek ve servis hizmetleri, gizlilik, genişletilmiş garantiler vb. katma değerli servisler de karşılaştırmaya dahil edilmektedir. Burada antlaşma tekliflere, karşı tekliflere ve kritiklere göre ve satışlar tüccarlar için yapılmaktadır.

#### 4.2.2 Minnesota ajan pazar yeri mimarisi (MAGMA)

Bu mimari, ticareti internette yürütmek için gerekli alt yapıyı, ajanlar arasında haberleşmeyi, otomatik ve insan kontrollü emirlerin gerçekleştirilmesini içerir. MAGMA'nın amacı, elektronik ticaretteki aktivitelerin yeterli ve tek biçimde tasarlanmasıdır. Alış ve satışı, banka işlemlerini, parasal işlemleri ve reklamları içerir. Kapalı zarf usulüne benzeyen fakat ikinci en düşük veya ikinci en yüksek fiyatı verenin kazandığı Vickrey mekanizmasına göre antlaşma yapılır.

#### 4.2.3 Firefly

Müzik albümleri ve sanatçılar için tavsiyelerde bulunan ağılı sistemdir. Sistem kullanıldıkça, kullanıcı ve sanatçı veritabanı gittikçe büyümektedir. Firefly'ın ana fikri:

- Kullanıcı profilini oluşturur, kullanıcının olumlu/olumsuz ilgilerini kaydeder,
- Oluşturduğu bu profili diğer kullanıcıların profilleriyle karşılaştırır. Diğer kullanıcıların bu profile benzerlik derecesini ölçer.
- Sonuç olarak, en benzer profilleri seçer ve kullanıcıya bu profiller içindeki malları tavsiye eder.

### 4.3 Antlaşma

Antlaşmayı anlayabilmek için öncelikle antlaşmayı ekonomik olarak tanımlamak gerekmektedir. Daha sonra antlaşmanın yapay zeka yönü incelenmiş, genetik programlama yöntemi verilmiştir. Son olarak arttırmanın basit tipleri ve çeşitleri aşağıda gösterilmiştir.

#### 4.3.1 Antlaşma sanat ve bilimi

Bilim ile yazar, problem çözmenin tanımının analizini kastetmektedir. Sanat ile kişiler arası yetenekleri, bir kişiyi ikna etme ve ikna olmayı ifade etmektedir. Antlaşma taraflar ve özellikler göz önünde bulundurularak sınıflandırılır. Buna göre dört çeşit antlaşma vardır:

- İki taraf tek kriter,
- İki taraf çok kriter,
- Çok taraf tek kriter,
- Çok taraf çok kriter.

İki taraflı antlaşma ikiye ayrılabilir: Dağıtılmış (tek özellik- fiyata göre) ve bütünleştirici (çok özellik).

Çok özellikte, hangi özelliğin değerine göre daha iyi olduğunu belirtmek için “**additive scoring system**” kullanılabilir. Bu sistemde, birkaç özellik belirlidir ve diğer özellikler sabit kabul edilir. Örneğin, inşaat firması alıcı arasında geçen uygulamada alıcı; sadece fiyata, teslim süresine ve kaliteye göre alım kararı versin. Alıcının verdiği alım emrine göre, satıcı firmanın 3-4.5 milyon dolar arasında fiyat, teslim süresini 250-400 gün, bina kalitesini 2 olarak verdiğini kabul edelim. En iyi kalite 1 ile en kötü kalite 5 ile ifade edilsin. Toplam ağırlık 1 olacak şekilde, önem derecesine göre faktörlere ağırlık atanır. 0 ile 100 arasında her bir faktöre puan verilir. Fiyat faktörüne 0.5, süre faktörüne 0.3 ve kalite faktörüne 0.2 ağırlık verildiğinde 4 milyon dolar, 350 gün ve 2nci derecede kalite istendiğinde ve her bir faktöre puan atandığında, toplam puan  $(0.5*50)+(0.3*25)+(0.2*80= 48.5)$  olarak bulunur. Elde edilen toplam puanlar arasında en yüksek olan toplam puan kazanır. Bu sistem değer puanlama sistemi (value scoring system) olarak adlandırılmaktadır.

### 4.3.2 Kendi kendini yürüten ajanlar için antlaşma kararı

Çeşitli endüstrilerde bir çok kendi kendini yürüten ajanlar alım kararını belirlemede kullanılmaktadır. Bu ajanlar bir malı alabilmek için birbirlerine teklif ve karşı tekliflerde bulunmaktadır. Sonuçta birden fazla partinin karar verdiği bir birleşik karar verme olarak düşünülmelidir. Antlaşmayı yürüten kendi kendini yürüten ajanları yazmak için, üç konu göz önünde bulundurulmalıdır: antlaşma protokolü, hangi konular üzerinde antlaşmanın olacağı, ajanın kullandığı çıkarım modeli.

### 4.3.3 Çoklu özelliklere göre antlaşmada yapay ajan öğrenme kuralları

Genetik algoritmalar ve genetik programlama, antlaşma konuları ile ilgilenen ajanlara birbiri ile nasıl antlaşma yapılacağını gösterir. Belirli bir eşik değerinde teklif verilir, eğer karşı tarafın eşik değerinde alım kararı verilmemişse karşı taraf tekrar teklifte bulunur. Bu teklif, karşı teklif verme işlemi belirli bir sayıya kadar tekrar eder. Ortak bir paydada çözüme ulaşılır. Daha sonra genetik algoritma çalıştırılır ve yeni stratejiler yaratılır.

### 4.3.4 Çok ajanlı sistemlerde artırma ile satış

Artırma teorisi protokolleri ve ajan stratejilerini artırma için analiz ederler. Burada bir artırmacı ve birden fazla teklif veren bulunmaktadır. Her mal için ayrı bir artırma şekli olmalıdır. Aşağıda başlıca dört çeşit artırma kuralı verilmiştir.

- İngiliz (ilk fiyat açık artırma) arttırmasına göre, en yüksek fiyatı veren kazanır. Burada ajanın stratejisi eski verilen fiyatlara göre belirlenir. Yani diğer arttırmacıların verdikleri fiyatı tahmin ederek, kendisi için uygun bulduğu en fazla fiyata kadar mücadele edilir.
- Tek fiyat kapalı usule göre, diğerlerinin verdiği teklif bilinmeden kapalı zarf usulü ile tek bir fiyat verilir. Diğer arttırmacıların verdikleri bütün fiyatlara ve kendisinin verebileceği maksimum fiyat bilgilerini önceden bilerek ajan, stratejisini belirler.
- Açık eksiltme.
- Vickrey.

## 5 WEB SERVİSLERİ

Web servisleri Haziran 2000'de ortaya çıkan bir çok yazılım firması tarafından yoğun ilgi ve destek bulan bir modeldir. Bu model, şu anda gelişme aşamasındadır. Bu aşamada web servisleri ile kabul edilen standartlar SOAP, WSDL ve UDDI'dir. İnternet üzerinde uygulama bütünleştirmesi için yöntem arayan bir çok yazılım geliştirici firma ve kurum, web servisleri modeli konusuna yönelmişlerdir. Bu modeli destekleyen Microsoft, IBM, Sun, HP, Oracle ve daha bir çok firma bu konuda yoğun bir şekilde çalışmakta ve web servisleri yazılım ve uygulama geliştirme araçları sunmaktadır. Bu konudaki firmaların yoğun desteğinden dolayı, uygulama bütünleştirmesi konusunda ortaya çıkacak hakim ortamın web servisleri modeline dayanacağı öngörülmektedir. Bu bölümde çok yeni olan web servisleri modeli açıklanmıştır.

Geçen birkaç yılda XML, web üzerinde heterojen hesaplama ortamları arasında bilgiyi paylaşmamızı sağlamıştır. Günümüzde ise XML, web servisleri kavramı sayesinde, sadece verinin yapısal olarak gösterimini sağlamamakta, ayrıca uygulamalar arasında mesajlaşmanın yapısal olarak gösterimini de sağlamaktadır. Bu da dağıtılmış programlamada, büyük bir ilerlemedir (11).

Bu bilgiden yola çıkarak, web servislerini şöyle tanımlayabiliriz (Glass, 2001):

**“Tek bir varlık olarak paketlenen ve diğer programlar tarafından kullanılmak üzere ağa yayımlanan bir dizi fonksiyondur. Web servisleri açık dağıtılmış sistemlerin temel taşıdır ve kurumların ve kişilerin hızlı ve ucuz olarak dünya üzerinde sayısal kar elde etmesini sağlamaktadır.”**

Web servisleri, gerçekte; kurumlara, uygulamalara ve sistem servislerine XML-tabanlı arayüzlerdir. Bir web servisini gerçekleştirmek için aşağıdaki adımlar izlenir:

- Sağlayıcı, kendi isteğine bağlı olarak bir programlama dili, middleware ve platformunu kullanarak web servisini yaratır, birleştirir ve internette yayınlar.
- Sağlayıcı web servisini, WSDL olarak tanımlar. Böylece web servisi, diğer kurumlara ve kişilere tanıtılmış olur.



- Sağlayıcı web servisini, UDDI'ye kaydeder. UDDI, web servislerinin yayınlanmasını ve diğer kişiler tarafından sağlanan servisleri aramamıza yarayan bir yazılım sağlamaktadır.
- Olası kullanıcı, UDDI kayıtlarını aratarak aradığı servisi bulur.
- Kullanıcının uygulaması, web servisini bağlar (bind) ve servisin işlemlerini SOAP kullanarak çağırır. SOAP, parametreleri XML formatında tutmamıza ve değerlerin HTTP üzerinden döndürülmesini sağlamaktadır. Bu, web servislerinin kullandığı haberleşme protokolüdür.

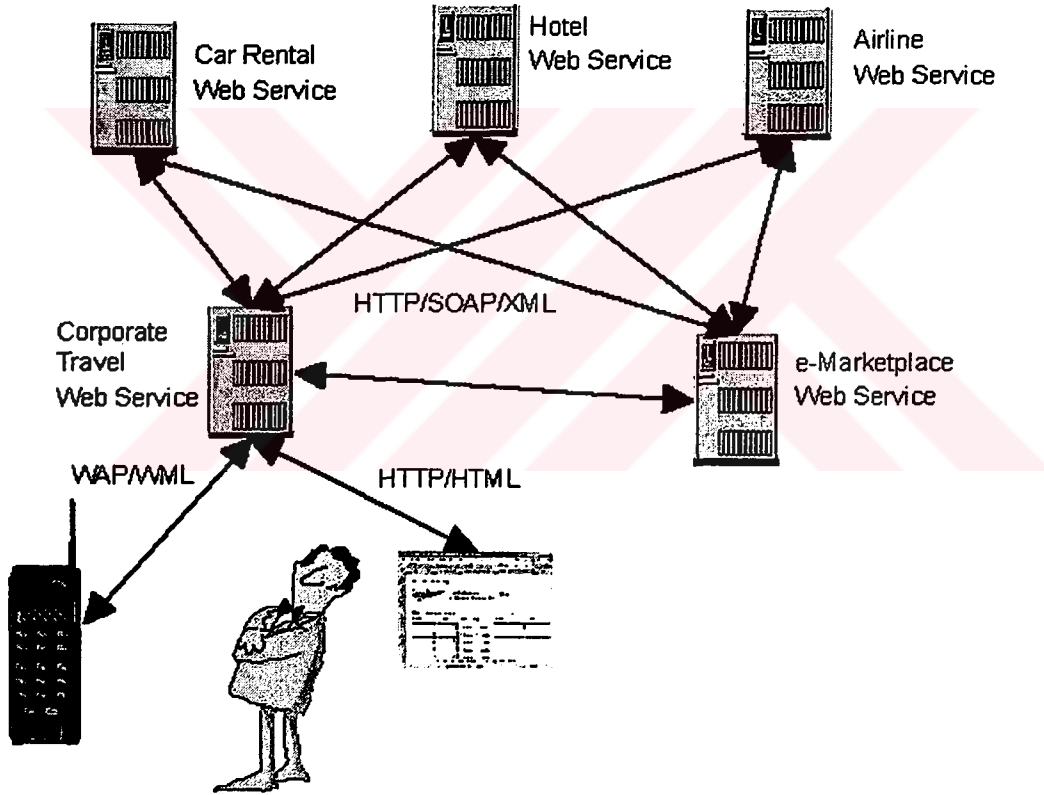
Yukarıdaki işlemler, basit bir web servisini yaratmak için yeterlidir. Karmaşık web servislerini tanımladığımızda, iş ve şemaların yapısı hakkında satıcı kurumların bir standart oluşturmaları gerekmektedir. Günümüzde satıcı kurumlar arasında tek bir standart oluşturulamamıştır fakat SOAP, WSDL ve UDDI'nin iyi olduğu konusunda satıcı kurumlar (Sun, Microsoft, IBM, Oracle, BEA) hem fikirdir (11). Web servislerinin tanımlamaları yapılmadan önce, web ortamındaki gelişmeler aşağıda verilmiştir.

## 5.1 Web Ortamındaki Gelişmeler

Geçen altı yıldaki gelişmelerden sonra, bugünkü web ortamı olmadan ağ tabanlı bilgi sistemlerinin düşünülmesi çok zordur. Kolay kullanılabilir ve ulaşılabilir olması webin başarısında en büyük etkidir. Web ortamındaki gelişmeleri üç safhada inceleyebiliriz (Akyokuş, 2001):

- *Belge Web'i (Document Web)*: Belge web'i ile web ilk aşamadaki kullanımı yani HTTP protokolü ile HTML dilinde biçimlendirilmiş **statik belgelerinin** kullanıcılara sunumunu ifade etmekteydi.
- *Uygulama Web'i (Application Web)*: Bu yapı kurumların müşterilerine web üzerinden bazı iş süreçlerini yaptırma gereksinimi sonucunda ortaya çıktı. Bu yapıda sunucu tarafında çalışan programlar (server-side programs) vasıtasıyla hazırlanan **dinamik HTML belgeleri** ile kullanıcı ve iş uygulaması arasında etkileşim sağlandı.
- *Servis Web'i (Services Web)*: Kurumların diğer kurumlarla olan iş süreçlerini bütünleştirme gereksinimi sonucunda ortaya çıkan ve gelişmekte olan yeni yapıdır. Bu yapının temel taşı web servisleridir. Web servislerindeki temel amaç kurum bilgi

sistemlerindeki **program modüllerinin etkileşimini** sağlamaktır. Web servisleri web ortamında yayınlanabilen, aranıp bulunabilen ve çağrılarak erişilebilen modüler uygulama fonksiyonlarıdır. Bu fonksiyonlar, değişik kurumsal iş süreçlerini gerçekleştireceklerdir. Web servisi kavramı destekçilerinden Frank Boss'a göre "İleride internette web sayfasından daha çok web servisi bulunacaktır." Şekil 5.1'de, web servislerini kullanarak bir iş gezisi için gerekli rezervasyonları yapan bir uygulama örneği senaryosunu göstermektedir. Bu örnekte iş gezisi planlayan kişinin; hava yolu, otel ve araç kiralama rezervasyonları yapacağı varsayılmıştır. İş gezisi organizasyonu için geliştirilen uygulama; havayolu, otel, araç kiralama ve e-marketplace kurumlarının sağladığı web servislerini çağırarak gerekli işlemleri tek bir uygulama ile gerçekleştirecektir.

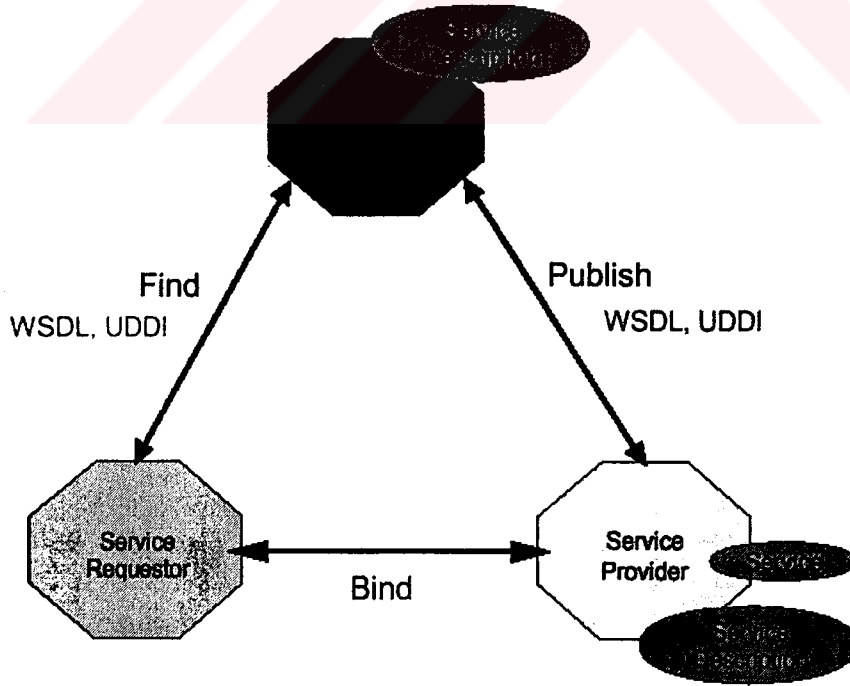


Şekil 5.1 Web servisleri ile etkileşim sağlayan iş gezisi uygulaması (Kaynak: IBM)

## 5.2 Web Servisleri Modeli

Web servisleri modeli üç ana birimin etkileşimine dayanır (Şekil 5.2). Bu üç ana birim şunlardır:

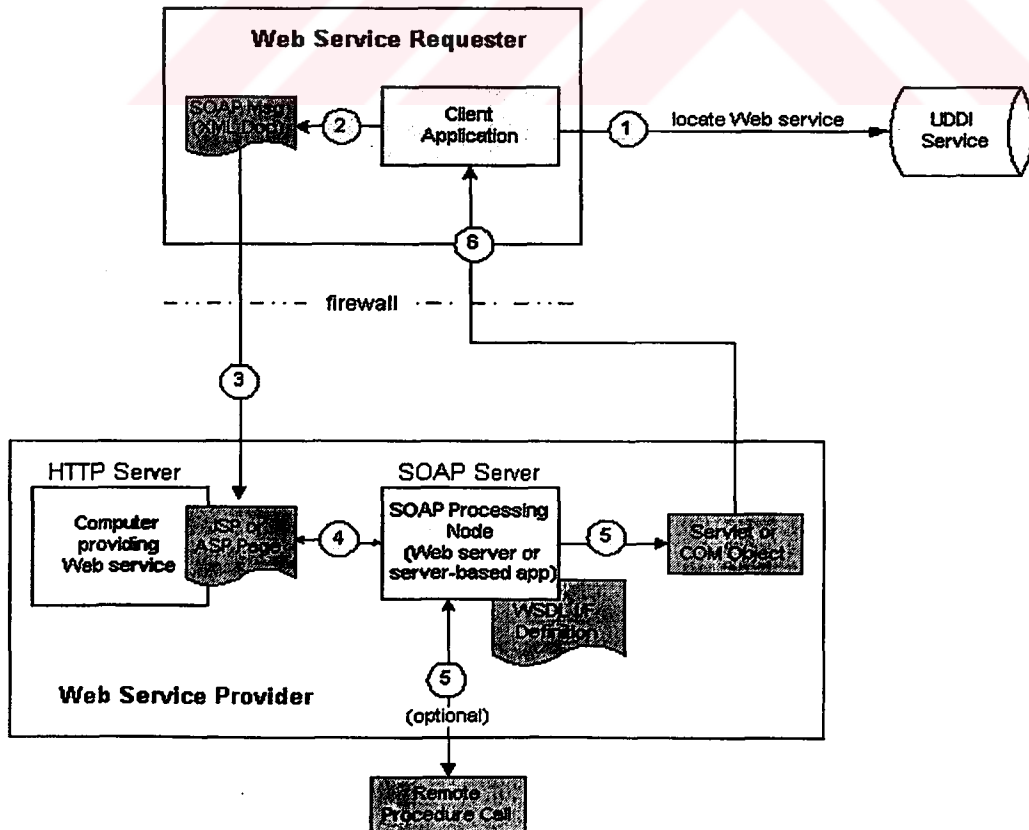
- **Servis Sağlayıcı (Service Provider):** Servis sağlayıcı, istemcilerin sağlayıcıda bulunan servislere erişimini sağlar. Servis sağlayıcı, sitesinde bulunan web servisleri tanımını servis kayıt birimine (service registry) kaydederek bu servisin nasıl çağrılacağını belirtir.
- **Servis İstemcisi (Service Requester):** Servis sağlayıcısında bulunan web servislerini çağırarak kullanan istemci uygulamalardır. Web servisinin nasıl çağrılacağı ve ilgili parametreleri servis kayıt biriminden arayarak bulur ve çağırır.
- **Servis Kayıt Birimi (Service Registry):** Servis sağlayıcılarının yayınladıkları web servisi tanımlarını saklar ve aranıp bulunmasını sağlar. Servis sağlayıcıları servis kayıt birimini tarayarak istediği servisler hakkında bilgi alabilir. Servis kayıt birimi her servisin nasıl çağrılacağı konusunda tanım bilgileri içerir.



Şekil 5.2 Web servis modeli (Kaynak: IBM)

Şekil 5.3'teki şema, bir web servisi istemcisinin bir servis sağlayıcıdan bir servisi çağırma aşamasındaki temel adımları göstermektedir. Şekil 5.3'e göre:

1. Web servisi istemcisi (SOAP Client), servis kayıt biriminden (UDDI) web servisini bulur.
2. İstemci bir SOAP mesajı hazırlar. SOAP mesajı bir XML belgesidir.
3. İstemci, SOAP mesajını web server veya uygulama sunucusunda çalışan SOAP istek dinleyicisine gönderir. İstek dinleyici, gelen isteklere cevap veren sunucu programlardır. Bu programlar bir JSP, ASP, CGI veya ISAPI programıdır.
4. SOAP sunucu, gelen SOAP mesajını çözümler ve gerekli parametreleri göndererek istenen nesnenin istenen yöntemini çağırır.
5. Çağrılan nesnedeki yöntem çalışır ve sonuçları SOAP sunucusuna gönderir. SOAP sunucusu, gelen sonucu SOAP mesajı formatında biçimlendirerek istemciye gönderir.
6. İstemci, gelen SOAP mesajının içindeki bilgileri alarak istekte bulunan programa gönderir.

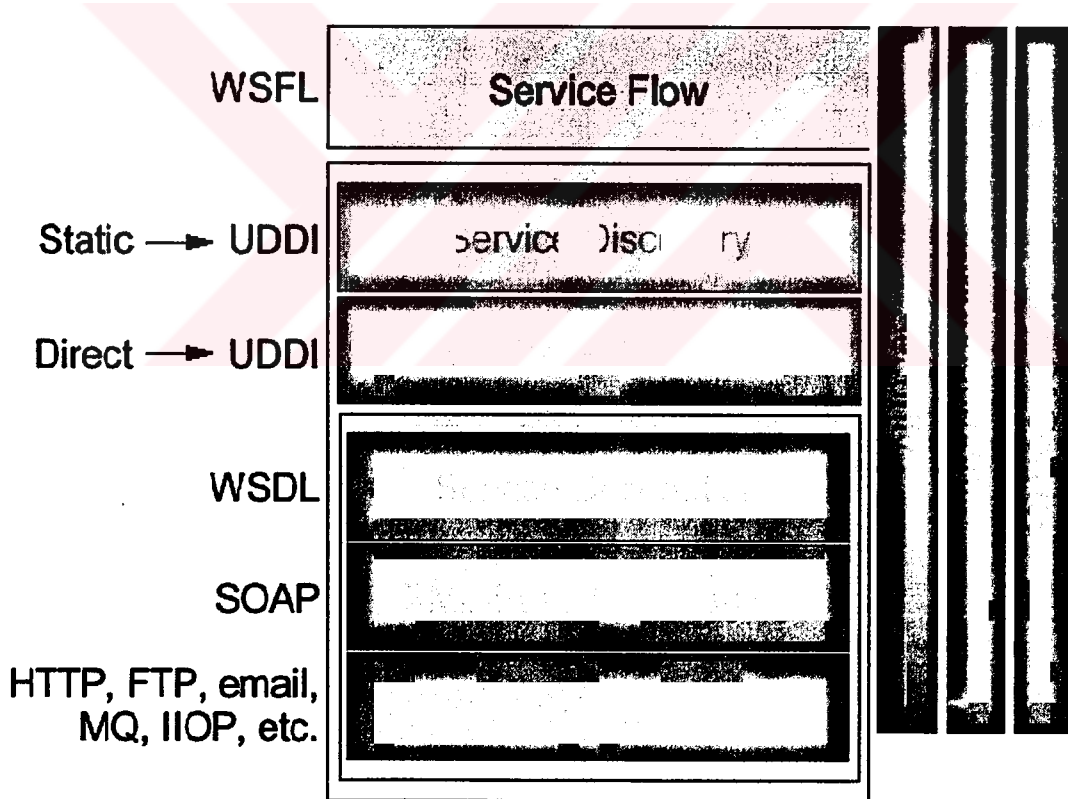


Şekil 5.3 Web servisi istemci ve sağlayıcısı arasındaki temel işlemler (Kaynak: Sun)

### 5.3 Web Servisleri Standartları

Web servisleri açık internet standartlarına dayanır. Daha önce belirtildiği gibi web servisi modeli Haziran 2000’de ortaya çıkmıştır. Bu nedenle henüz tamamen olgunlaşmış teknoloji değildir. Şekil 5.4, web servisi mimarisindeki temel katmanları göstermektedir. Bu katmanlarda belirtilen güvenlik, iş akışı, servis kalitesi ve yönetim gibi konulardaki web servisi standartları henüz araştırma aşamasındadır. Web servisleri aşağıdaki çekirdek standartlardan oluşmaktadır (Akyokuş, 2001).

- SOAP: İnternet üzerinde web servislerini çalıştırmak için kullanılan protokol,
- WSDL: Web servislerini tanımlama dili,
- UDDI: Web Servislerini kaydeden, bulan ve kullanan merkezi organizasyon.



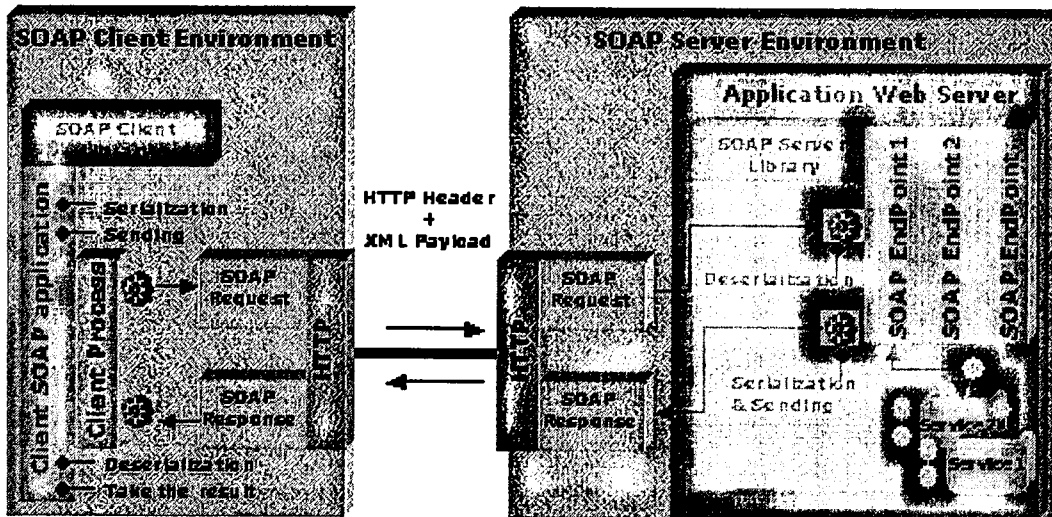
Şekil 5.4 Web servisi mimarisi katmanları (Kaynak: IBM)

### 5.3.1 SOAP

SOAP, uygulama bütünleştirmesi için geliştirilmiş XML ile formatlanmış bilgilerin iletişimini sağlayan bir protokoldür. Uygulama bütünleştirme için, *middleware* olarak adlandırılan daha önce bir çok çözümler sunulmuştur. RPC (Remote Procedure Call, DCOM, IIOP (Internet Inter-ORB Protocol) ve Java RMI bu çözümlerden bazılarıdır. Bu middleware çözümleri, internet ortamında iletişim sağlama konusunda yetersiz kalmaktadır. Bu nedenle XML tabanlı bir protokol olan SOAP giderek yaygınlaşmaktadır. SOAP istemcilerin, sunucularda olan nesne yöntemlerini çağırmasını ve sonuçların alınmasını sağlayan basit istek/yanıt (request/response) protokolüdür.

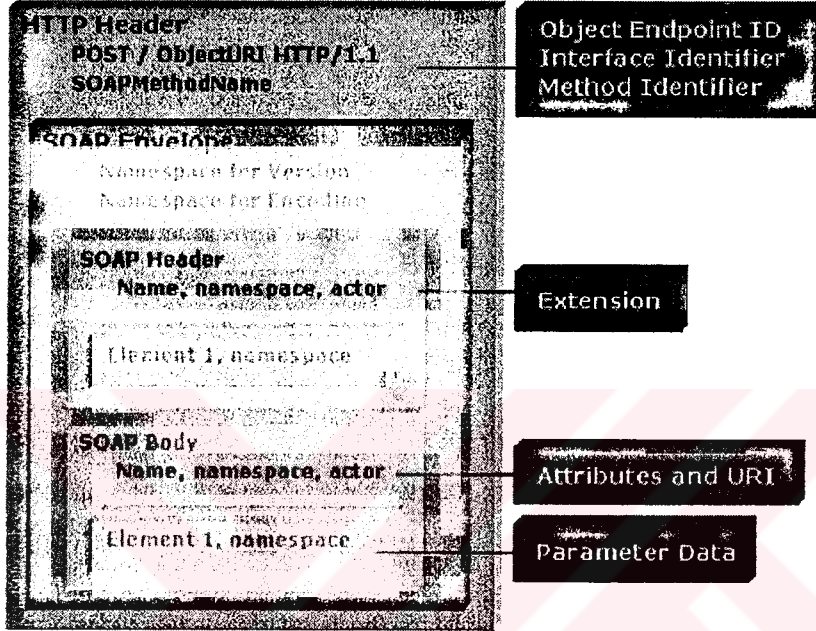
SOAP, mevcut internet altyapısında olan router, firewall ve proxy sunucularda herhangi bir değişiklik yapmadan kolayca çalışmaktadır.

Şekil 5.5 bir SOAP istemci ve sunucusu arasındaki iletişimi göstermektedir. Bir SOAP uygulaması geliştirmek için, istemci ve sunucuya SOAP geliştirme araçları ile birlikte gelen kütüphanelerin yüklenmesi gerekir. Bu kütüphaneler, bir XML çözümleyici ve SOAP işlemcisi içerir. İstemci SOAP uygulaması, bir SOAP istek mesajı oluşturarak bu isteği SOAP sunucusunda tanımlanmış servis uç noktalarından (end point) birisi tarafından çalıştırılması için gönderir. SOAP sunucu, ilgili servisi çalıştırdıktan sonra SOAP yanıt mesajı hazırlar. Hazırlanan SOAP yanıt mesajı istemciye iletilir.



Şekil 5.5 SOAP istemci-sunucu arasındaki iletişim (Kaynak: www.techmetrix.com)

Şekil 5.6, HTTP protokolü ile gönderilen bir SOAP mesajını göstermektedir. SOAP mesajı, HTTP POST metodu veri paketinin içinde gönderilir. Bir SOAP mesajı, bir SOAP zarfından (SOAP envelope) oluşur. SOAP zarfı seçmeli bir SOAP başlığı (SOAP header) ve SOAP gövdesinden (SOAP body) oluşur. SOAP gövdesi çağrılacak metod ve metodun içerdiği parametreleri içerir.



Şekil 5.6 Bir SOAP mesajının yapısı (Kaynak: www.techmetrix.com)

```
POST /StockQuote HTTP/1.1
Host: www.stockquoteserver.com
Content-Type: text/xml; charset="utf-8"
Content-Length: nnnn
SOAPAction: "http://example.com/stockquote.xsd"

<SOAP-ENV:Envelope
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
  <SOAP-ENV:Body>
    <m:GetLastTradePrice xmlns:m="http://example.com/stockquote.xsd">
      <symbol>DIS</symbol>
    </m:GetLastTradePrice>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Şekil 5.7 Bir SOAP istemci istek (request) mesajı

```

HTTP/1.1 200 OK
Content-Type: text/xml; charset="utf-8"
Content-Length: nnnn

<SOAP-ENV:Envelope
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
  <SOAP-ENV:Body>
    <m:GetLastTradePriceResponse xmlns:m="http://example.com/stockquote.xsd">
      <Price>34.5</Price>
    </m:GetLastTradePriceResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

Şekil 5.8 Bir SOAP yanıt (response) mesajı

### 5.3.2 WSDL

Bir uygulamanın bir web servisini kullanması için web servisinin nasıl çağırılacağı, arayüzünün, hangi protokolleri kullandığının ve kodlama standartlarının belirtilmesi gerekir. WSDL web servisini tanımlayan bir XML belgesidir. Bir anlamda dağıtık programlamada kullanılan IDL (Interface Definition Language – Arayüz Tanımlama Dili)’e benzer. Web servisi tanımlama işlemleri, giren ve çıkan mesaj formatları, ağ ve port adresleri gibi bilgileri tanımlar. Bir web servisi tanımlama belgesi aşağıdaki temel elemanları içerir:

- *Types*: mesajlarda kullanılacak veri tiplerini belirtir.
- *Message*: İletişimde kullanılacak mesajları tanımlar.
- *PortType* : Web servisinin içerdiği işlemleri (methods) ve ilgili mesajları tanımlar.
- *Binding* : İşlem ve mesajlarda kullanılacak veri formatlarını tanımlar.
- *Port*: Binding ve web adresinden oluşan servis noktasını tanımlar. Web adresi servisin çalıştırılacağı URL’dir.
- *Service*: Kullanılan port’lar kümesidir.

Şekil 5.9, Microsoft’un web servisleri aracı olan Microsoft SOAP toolkit’te verilen bir basit aritmetik işlemler yapan bir web servisinin WSDL belgesini göstermektedir. Bu web servisi; EchoString, AddNumbers ve SubtractNumbers işlemlerini gerçekleştirmektedir. Şekil 5.9’da görüldüğü gibi WSDL belgesi oldukça karmaşıktır. Bu nedenle genellikle web servisi geliştirme araçları, WSDL belgelerini otomatik olarak oluşturan programlar içermektedir.



```

<definitions name='DocSample1' targetNamespace='http://tempuri.org/wsd1'
  xmlns:wsd1ns='http://tempuri.org/wsd1'
  xmlns:typens='http://tempuri.org/type'
  xmlns:soap='http://schemas.xmlsoap.org/wsd1/soap'
  xmlns:xsd='http://www.w3.org/2001/XMLSchema'
  xmlns:stk='http://schemas.microsoft.com/soap-toolkit/wsd1-extension'
  xmlns='http://schemas.xmlsoap.org/wsd1/'>
  <types>
    <schema targetNamespace='http://tempuri.org/type'
      xmlns='http://www.w3.org/2001/XMLSchema'
      xmlns:SOAP-ENC='http://schemas.xmlsoap.org/soap/encoding'
      xmlns:wsd1='http://schemas.xmlsoap.org/wsd1/'>
    </schema>
  </types>
  <message name='Sample1.EchoString'>
    <part name='testString' type='xsd:string' />
  </message>
  <message name='Sample1.EchoStringResponse'>
    <part name='Result' type='xsd:string' />
  </message>
  <message name='Sample1.AddNumbers'>
    <part name='NumberOne' type='xsd:double' />
    <part name='NumberTwo' type='xsd:double' />
  </message>
  <message name='Sample1.AddNumbersResponse'>
    <part name='Result' type='xsd:double' />
  </message>
  <message name='Sample1.SubtractNumbers'>
    <part name='NumberOne' type='xsd:double' />
    <part name='NumberTwo' type='xsd:double' />
  </message>
  <message name='Sample1.SubtractNumbersResponse'>
    <part name='Result' type='xsd:double' />
  </message>
  <portType name='Sample1 SoapPort'>
    <operation name='EchoString' parameterOrder='testString'>
      <input message='wsd1ns:Sample1.EchoString' />
      <output message='wsd1ns:Sample1.EchoStringResponse' />
    </operation>
    <operation name='AddNumbers' parameterOrder='NumberOne NumberTwo'>
      <input message='wsd1ns:Sample1.AddNumbers' />
      <output message='wsd1ns:Sample1.AddNumbersResponse' />
    </operation>
    <operation name='SubtractNumbers' parameterOrder='NumberOne NumberTwo'>
      <input message='wsd1ns:Sample1.SubtractNumbers' />
      <output message='wsd1ns:Sample1.SubtractNumbersResponse' />
    </operation>
  </portType>
  <binding name='Sample1 SoapBinding' type='wsd1ns:Sample1 SoapPort' >
    <stk:binding preferredEncoding='UTF-8' />
    <soap:binding style='rpc' transport='http://schemas.xmlsoap.org/soap/http' />
    <operation name='EchoString' >
      <soap:operation soapAction='http://tempuri.org/action/Sample1.EchoString' />
      <input>
        <soap:body use='encoded' namespace='http://tempuri.org/message'
          encodingStyle='http://schemas.xmlsoap.org/soap/encoding' />
      </input>
      <output>
        <soap:body use='encoded' namespace='http://tempuri.org/message'
          encodingStyle='http://schemas.xmlsoap.org/soap/encoding' />
      </output>
    </operation>

```

```

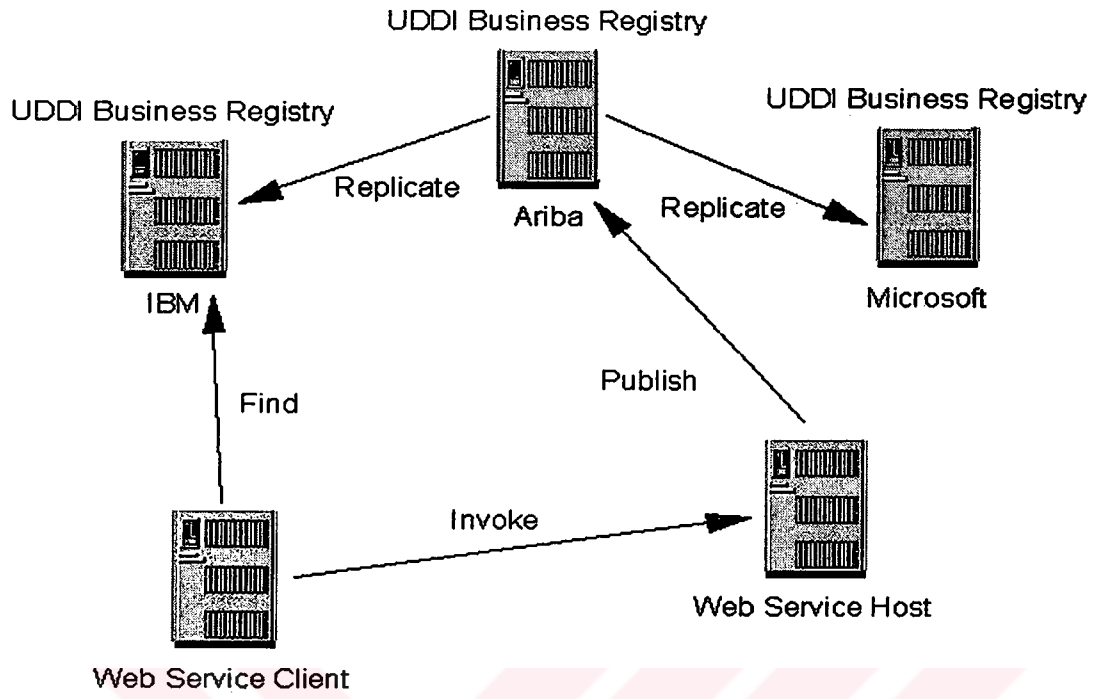
<operation name='AddNumbers' >
  <soap:operation soapAction='http://tempuri.org/action/Sample1.AddNumbers' />
  <input>
    <soap:body use='encoded' namespace='http://tempuri.org/message/'
      encodingStyle='http://schemas.xmlsoap.org/soap/encoding' />
  </input>
  <output>
    <soap:body use='encoded' namespace='http://tempuri.org/message/'
      encodingStyle='http://schemas.xmlsoap.org/soap/encoding' />
  </output>
</operation>
<operation name='SubtractNumbers' >
  <soap:operation soapAction='http://tempuri.org/action/Sample1.SubtractNumbers' />
  <input>
    <soap:body use='encoded' namespace='http://tempuri.org/message/'
      encodingStyle='http://schemas.xmlsoap.org/soap/encoding' />
  </input>
  <output>
    <soap:body use='encoded' namespace='http://tempuri.org/message/'
      encodingStyle='http://schemas.xmlsoap.org/soap/encoding' />
  </output>
</operation>
</binding>
<service name='DocSample1' >
  <port name='Sample1SoapPort' binding='wsdl:ns:Sample1SoapBinding' >
    <soap:address location='http://localhost/DocSample1/DocSample1.wsdl' />
  </port>
</service>
</definitions>

```

Şekil 5.9 Bir WSDL belgesi

### 5.3.3 UDDI

Bir web servisini kullanmak için kullanıcının, web servisi sağlayan kurumları ve bu kurumların verdikleri web servislerinin neler olduğunu bilmesi gerekir. UDDI kısaltmasında geçen *Evrensel*, *Tanım*, *Buluş* ve *Bütünleştirme* kelimelerinin ifade ettiği gibi UDDI; kurumların kendilerini, sağladıkları servisleri yayımlayarak tanımlamalarını ve bu bilgilerin daha sonra diğer kurumlarca taranıp bulunmasını sağlayan bir standarttır. UDDI Kurum Kayıt Servisi (UDDI Business Registry) kurum ve web servisleri bilgilerini saklayan sunuculardır. Bu sunucular, servis sağlayıcılarından gelen bilgileri kendi veritabanlarına kayıt ederek diğer kurumların erişimine açar. Şu anda etkin olarak çalışan kurum kayıt sunucuları *uddi.microsoft.com* ve *uddi.ibm.com*'dur. Şekil 5.10'da görüldüğü gibi bu sunucular kendilerine kayıt edilen bilgileri diğer sunuculara da kopyalayarak kolay ve hızlı bir şekilde erişilmesini sağlarlar. UDDI sunucuları; kurum ve servis kayıt, güncelleme ve tarama işlemlerini web servisleri (SOAP mesajları) ile gerçekleştirir.



Şekil 5.10 UDDI kurum kayıt sunucuları

## 6 MICROSOFT BIZTALK SERVER 2000

Microsoft BizTalk Server 2000, kurumlar arası elektronik ticaret için gerekli olan uygulama entegrasyonunu en iyi şekilde gerçekleştirmemize yarayan bir dizi araçtan ve sunucu yazılımından oluşmaktadır. Bu yazılım ile dördüncü bölümde açıklanan BizTalk yapısı gerçekleştirilmektedir. Bu bölümde, Microsoft Server 2000 araçlarının işlevleri açıklanacaktır.

Server araçları ile kurumlar arasında mesajlaşmayı sağlayan XML dokümanları, BizTalk Editör ile tasarlanabilmekte ve yaratılan şema WebDAV şema tutucuda saklanabilmektedir. Kurumlar birbirlerinin şemalarında belirtilen bilgileri aynı şekilde kullanmıyorlarsa, bu bilgiler arasında eşleme yapılmaktadır. Bu eşleme, BizTalk Mapper ile gerçekleştirilebilmekte ve yaratılan eşleme şeması WebDAV'da saklanabilmektedir. BizTalk Mapper, ayrıca basit ve karmaşık veri işlemeyi de functoids paleti ile sağlamaktadır.

Tanımlanan şemalar kullanılarak, kurumlar arasında XML-tabanlı mesajlaşma gerçekleştirilmektedir. BizTalk Messaging Manager'da tanımlanan doküman, organizasyon, port, kanal mesajlaşma nesnelere ile mesajlaşma sağlanmaktadır.

Kurumlar arasında geçen bir işi modellemek için, tanımlanan mesajlaşma nesnelere belirli bir sırada çalışması gerekmektedir. Bir işi oluşturulan iş adımları, iş akışı olarak BizTalk Orchestration Designer'da tanımlanabilmektedir. Bu araç, bir iş adımını gerçekleştiren uygulama parçasını da çalıştırabilmektedir. Tanımlanan iş akışı, XLANG dilinde kaydedilmektedir.

İş akışları, uygulama entegrasyon parçalarının (AIC) processMessage metodu veya **“dosya alım”** fonksiyonu ile etkinleştirilmektedir. BizTalk Server Administration aracı ile dosya alım fonksiyonları tanımlanmakta ve yönetilmektedir. BizTalk Server Administration aracı, organizasyonlar arasında doküman gönderimini de yönetmektedir.

Bu doğrultuda, bir iş sürecini tanımlayabilmek için BizTalk Server'ın en azından üç görevi yapması gerekmektedir. Bunlar:

- İş Akışını tanımlama,
- İş Akışı sürecince mesaj akışlarını kontrol etme,
- İş Akışını izleme.

Sadece mesajları düşündüğümüzde ise; BizTalk Server, mesajların şeklini ve akışını kontrol eden ve aşağıdaki işlevleri sağlayan araçları içermelidir.

- Mesaj özelliklerini tanımlama,
- Bir mesajdan diğerine çevrimi tanımlama,
- Mesaj değişimini ve çevrimini kontrol etme,
- Mesaj içeriğini izleme.

## **6.1 BizTalk Mesajlaşma Servisleri**

BizTalk Server tarafından gerçekleştirilen düşük-seviyeli mesajlaşma servisleri: örneğin mesaj gönderme, alma, çözümlenme ve izleme; ve veri eşleme BizTalk Mesajlaşma Servisleri olarak adlandırılmaktadır. Burada öncelikle, mesajlaşma ile ilgili birkaç BizTalk-Server'a özel tanımın verilmesi gerekmektedir.

### **6.1.1 Organizasyonlar**

Doküman alışverişinde bulunduğunuz ticari ortakları belirler. Bir organizasyon, kurumların departmanları da olabilir.

### 6.1.2 Dokümanlar

BizTalk Server tarafından işlenen mesaj dokümanının içeriğini belirler. Gelen veya giden dokümanları gösteren işaretçilerdir. Mesaj içeriğinin hangi bilgilerden oluştuğunu ve tiplerini tutar.

### 6.1.3 Zarflar

Zarflar, gerçek hayattaki posta zarflarına benzemektedir. Ticari ortağa bir doküman gönderileceği zaman dokümanın içinde olacağı zarf da belirlenmektedir. BizTalk Server, dokümanın tipine göre bir zarfı kendi belirlemektedir. BizTalk Server, dokümanın kök elemanına bakarak doküman tipini anlamaktadır.

### 6.1.4 Portlar ve Dağıtım Listeleri

Doküman değişimini veya uygulamaya veya organizasyona gönderilmesini belirleyen özellik kümesidir. Messaging ports, transport servislerini, mesajın gideceği organizasyonu ve çalışacak uygulamayı, güvenlik ayarlarını ve zarf ayarlarını içermektedir. Dağıtım listeleri ise birden fazla porta bir isim verme ile elde edilmektedir. Birden fazla porta aynı mesaj gönderilecek ise, dağıtım listeleri kullanılmaktadır.

### 6.1.5 Kanallar

Dokümanları işlemek için BizTalk Server'ı uygun adımlara yönlendiren özellik kümesidir. Kanal, kaynak organizasyon veya uygulamayı, bir doküman tanımını, bir eşlemeyi (map) içermektedir. Her portun, bir kanalı vardır. Kanal, kaynak organizasyonundan mesajı alır ve eğer gelen dokümanın başka bir XML formuna çevrimi gerekiyorsa çevirir. Daha sonra kanalın bağlı olduğu porttaki uygulama çalıştırılır. Böylece portta tanımlı, mesajın gideceği organizasyona mesaj gönderilmiş olur.

## 6.2 Mesaj Transferi

Microsoft BizTalk Server 2000, uygulamalar arasında mesaj deęişimi için iki teknik kullanmaktadır:

- Açık internet protokolleri
- Windows'a özel haberleşme

### 6.2.1 Açık internet protokolleri

Web, uygulamaları açık haberleşme protokollerine dayanmaktadır. İnternet uygulamaları geliştirenler bu açık haberleşme protokollerini destekleyen uygulamalar gerçekleştirmelidir. Bu yüzden Microsoft BizTalk Server 2000, haberleşme için aşağıdaki protokolleri desteklemektedir.

- Simple Mail Transport Protocol (SMTP): e-mail ile haberleşme,
- Hypertext Transfer Protocol (HTTP): web sayfaları ve uygulamaları (ASP, CGI ve diğerleri) tarafından kullanılan haberleşme mekanizması,
- Secure Hypertext Transfer Protocols (HTTPS): istemci ve sunucu arasındaki trafięi şifreleyen Secure Socket Layer'larını kullanan HTTP versiyonu.

### 6.2.2 Windows'a özel haberleşme

BizTalk aşağıdaki Windows'a özel protokolleri içermektedir:

- Microsoft Message Queue
- Yerel dosya sisteminden dosyaları okuma ve yazma desteęi (dosya deęişimi)
- COM+

MSMQ ile başka makineye birçok mesajı güvenli olarak gönderebilmektedir. COM+, Windows'un çekirdek katmanıdır. Bu yüzden Microsoft BizTalk Server 2000 bu yazılım öęesini desteklemektedir. Dokümanlar, Interchange COM+ arayüzünün Submit() ve

SubmitSync() metotları ile BizTalk Server' a gönderilmektedir. Doğrudan çağrım yapabiliyorsa BizTalk Server, bu yöntem ile dokümanların gönderimi sağlanmaktadır. Eğer BizTalk doğrudan çağrım yapamıyorsa, alım fonksiyonları sayesinde bu dokümanlar gönderilmektedir. Alım fonksiyonları, sunucunun belirli dizinlerini dinlerler. Herhangi bir XML dosyası bu dizine yazıldığında, hemen doküman BizTalk Server'a verilmektedir.

BizTalk Server, veriyi uygulamaya göndermek için uygulama entegrasyon parçaları (Application Integration Components) olarak bilinen COM+ yazılım parçalarını kullanmaktadır. AIC ile doküman doğrudan uygulamaya verilmemektedir. Dokümanın gönderimi AIC ile yapılmaktadır. AIC üzerinden BizTalk'un haberleşme yapabilmesi için AIC'nin aşağıdaki iki arayüzden bir metodu kullanması gerekmektedir.

- IBTSAppIntegration
- IPipelineComponent

IBTSAppIntegration tek bir metodu içermektedir: ProcessMessage(). Bu metot ile buraya gelen doküman işlenmekte ve tekrar BizTalk Server'a verilmektedir.

### 6.3 BizTalk Server Araçları

- **Orchestration Designer:** İş Akışlarını tanımlayan görsel bir araçtır. Tanımlanan iş akışı XLANG dilinde kaydedilir. Ekranın sol tarafında iş süreci, sağ tarafında ise tasarım şekli görülür. Bu tasarım sayesinde, iş akışı sadece 1 defa tanımlanır. Tasarım detayı kolaylıkla değiştirilebilir.
- **Editör:** Mesaj şekillerini tanımlamak için kullanılır. Editör veritabanı terminolojisinden etkilenmiştir. XML-tabanlı özellikleri yazmak için, programcı recordları ve alanları temel XML veri tipleri olarak tanımlar.
- **Mapper:** Grafikselsel bir editördür. 2 mesajın bir diğerine çevrimi burada tanımlanır. Ayrıca Mapper, veri işleme özellikleri de sunar. Karmaşık veri işlemeyi sağlayan Functoids Palette'i vardır.
- **Messaging Manager:** Grafikselsel mesaj yönetim aracıdır. Kurumlar arası doküman değişimini sağlar.



- **Document Tracking:** Browser-hosted bir arayüzdür. Mesajlaşma servislerinden geçen mesajlar hakkında rapor sunumunu sağlar.
- **Server Administration:** Microsoft Management Console'dur. Bu araç sayesinde; mesaj kuyukları, alım fonksiyonları, sunucu özellikleri yönetilmektedir.

### 6.3.1 Editör

Bu editör mesaj özelliklerini belirlemek için kullanılmaktadır. BizTalk ile uygulamaların uyumluluğunu sağlamak için, yapısal mesajlaşmanın gerçekleştirilmesi gerekmektedir. Bu nedenle, mesaj içeriklerini XML formu ile gösteren ve bu XML dokümanını WebDAV'da bir şema olarak tanımlamamızı sağlayan BizTalk Editör kullanılmaktadır.

BizTalk Editör, veritabanı terminolojisinden etkilenmiştir. XML-tabanlı özellikleri yazmak için programcı, recordları ve alanları temel XML veri tipleri olarak tanımlar.

#### 6.3.1.1 WebDAV

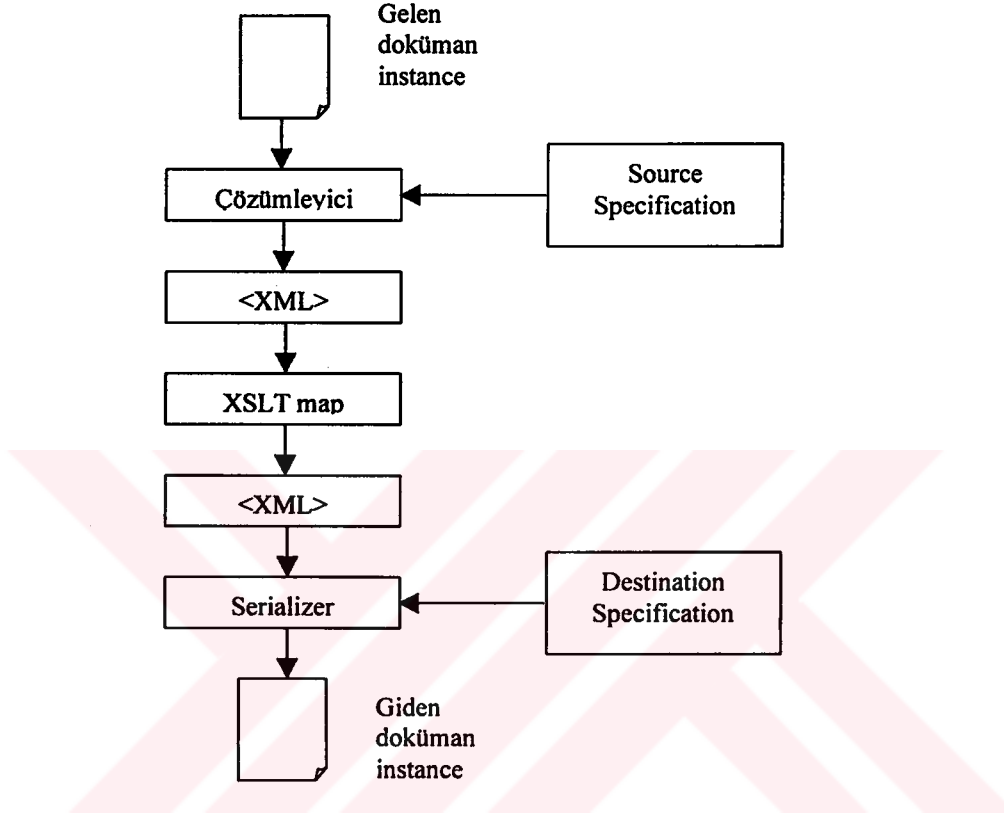
Web Distributed Authoring and Versioning (WebDAV), Web üzerinde ortaklaşa doküman düzenlemesi için Internet Engineering Task Force (IETF)'nin standartıdır. Dokümanı okuyup üzerinde çalışmaya olanak sağlar.

WebDAV sayesinde dokümanlara web üzerinden erişilebilir. WebDAV ayrıca dokümanları kilitleme özelliğine de sahiptir. Böylece bir kullanıcının yapmak üzere olduğu değişiklikler boşa gitmez.

### 6.3.2 Mapper

Mapper, bir grafiksel editördür. 2 mesaj özelliğini aynı anda gösterir. Farklı şemaları eşlemek için kullanılmaktadır. Ayrıca veri işleme özellikleri de sunar. Bu, Functoids ile sağlanır. Karmaşık veri işleme, functoids'de tanımlı scriptler sayesinde de yapılabilir.

İstenen eşleme sağlandığında programcı, Mapper'ı kullanarak derler. Sonuçta XML transformasyonundan XSLT stylesheet elde edilir. Daha sonra mesaj özellikleri arasında eşleşme istendiğinde runtime sunucu süreci uygun mesaj eşlemesini uygulayarak uygun stylesheeti seçer. Bu işlem, Şekil 6.1'de verilmiştir.



Şekil 6.1 BizTalk Server'in doküman işleme süreci

### 6.3.3 Orchestration Designer

BizTalk Server'in amacı, uygulama entegrasyonudur ve uygulamaları birbiri ile entegre etmenin ilk nedeni: iş akışlarını tanımlamaktır. Bu araç iş akışlarını tanımlamak için kullanılmaktadır.

İş akışını tanımlama ve gerçekleştirme **orchestration** olarak bilinmektedir. BizTalk Server Orchestration Designer bu görevi yürütmektedir. BizTalk'ta sistemler, uygulamalar arasında mesajların değişimi sayesinde gerçekleştirilmektedir. Tanımlanacak sistemde her adımda hangi mesajın gönderileceği veya alınacağı belirlenir. Ayrıca mesajın hangi sunucudandan alınacağı veya gönderileceği tanımlanır.

Orchestration Designer, sistemin iş akışını göstermektedir. Bu iş akışıyla verinin nasıl bir formdan ötekine değiştirildiği görülebilmektedir. Daha sonra bu iş akışı, XLANG dilinde, Orchestration Designer ile kaydedilmektedir. Tanımlanan bu iş sürecinin parçalarının internet üzerinden çalışması için saatler geçebilmektedir. İş süreci tekrar çalışmaya başladığı zaman iş akışı “hydrate” edilmektedir. “XLANG Scheduler” beklediği zaman iş akışı “dehydrate” edilmektedir. İş akışının hangi adımının çalıştırıldığı, veritabanına kaydedilmektedir.

BizTalk Orchestration Designer ile çok uzun süre çalışan iş süreçleri tasarlanabilmekte, bu süreçteki her bir işlemin (action) gerçekleştirim özellikleri belirlenebilmekte ve bu bilgi çalışabilir XML gösterimi olarak (XLANG iş akışı) derlenebilmektedir. Bu iş akışları, zamana, organizasyonlara ve uygulamalara göre “loosely coupled” olarak dağıtılabilmektedir.

BizTalk Orchestration Designer, iş süreçlerini tasarlamamıza olanak sağladığı gibi bu tanımlanan iş adımlarını, önceden yazılmış uygulamalarla bütünleştirmemize de olanak sağlamaktadır. Bu nedenle, BizTalk ürünü diğer kurumlar arası elektronik ticaret yapılarından ve ürünlerinden ayrılmaktadır. Aşağıda BizTalk Orchestration Designer ile gerçekleştirilebilen gerçekleştirim teknolojileri verilmiştir.

### 6.3.3.1 Gerçekleştirim Teknolojileri

BizTalk Orchestration Tasarımcısı, 4 adet yazılım şekli sunmaktadır. Bu şekiller iş süreçlerinde portu tanımlamak için yazılım teknolojilerini tanımlamayı sağlar.

- **COM Components:** Önceden yazılmış yazılım parçalarını veya uygulamaları, iş akışında kullanılmasını sağlar. COM gerçekleştirim teknolojisi, eş zamanlıdır. İş gerçekleştiğinde iki yönlü mesaj akışı vardır.
- **Windows Script Components:** Windows Script Component’lerin (wsc), iş akışında kullanılmasını sağlar.. Bu teknoloji, eş zamanlıdır. İş gerçekleştiğinde iki yönlü mesaj akışı vardır.
- **Message Queueing:** Diğer iş akışı ile iletişim kurmayı sağlar.
- **BizTalk Messaging:** BizTalk Orchestration Servisleri ile BizTalk Messaging Servisleri arasında mesaj değişimini sağlamaktadır.

#### 6.3.4 Server Administration

BizTalk Server Administration ile BizTalk Server'a verilen işler ilgili kuyruklarda görülebilmekte, dosya alım fonksiyonları tanımlanmakta ve izlenmektedir. Gelen ve giden dokümanları yönlendirmek ve işlemek için aşağıdaki kuyruklar kullanılmaktadır:

- Çalışan işler kuyruğu (Work queue),
- Sunucuya verilen işler kuyruğu (Scheduled queue),
- Tekrar çalıştırılacak işler kuyruğu (Retry queue),
- Durdurulmuş işler kuyruğu (Suspended queue).

Böylece hata oranı da azaltılmaktadır. Doküman önce işler kuyruğuna gelmektedir. Buradan uygun olan sunucu, dokümanı almakta ve işlemektedir. Eğer sorun çıkarsa işlem bir daha denenmek üzere tekrar çalıştırılacak işler kuyruğuna atılmaktadır. Hiç bir şekilde doküman işlenemezse bu durumda doküman, durdurulmuş işler kuyruğuna gönderilmektedir.

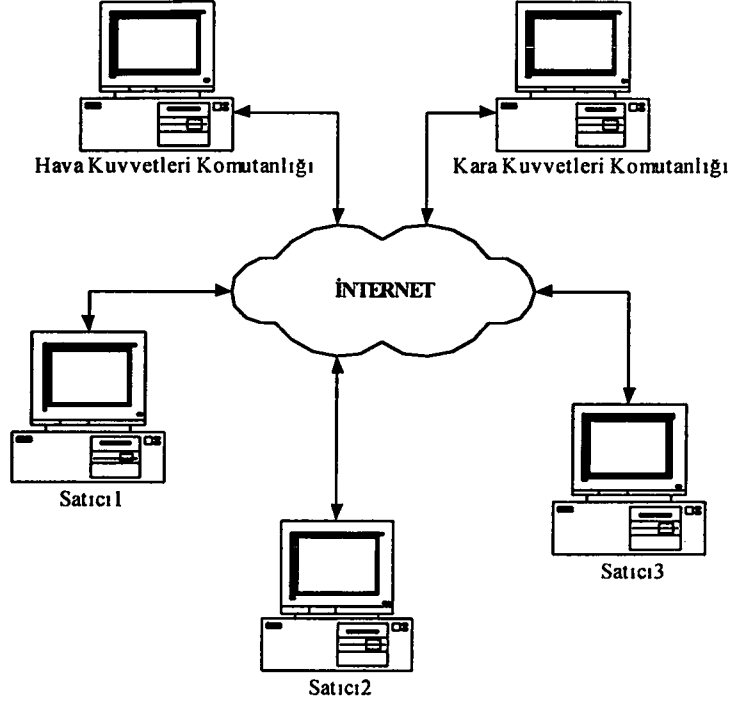
## 7 TASARIM

Basit bir kurumlar arası elektronik ticaret modeli oluşturmak, kurumlar arası uygulama geliştirmede ortaya çıkabilecek güçlükleri görmek, iş süreçlerini modellemek, mesajlaşma yapılarını tanımlamak, böyle bir uygulamanın analizi, tasarımını ve gerçekleştirimini yapmak, çok yeni olan kurumlar arası elektronik ticaret geliştirme araçlarını kullanmak ve bunların özelliklerini tanımak için kamu kurumu ile satıcı kurumlar arasında geçen kurumlar arası elektronik ticaret modeli bu bölümde tasarlanacaktır.

Kurumlar arası ticaret modelimizde iki alıcı kurum olarak, Hava Kuvvetleri Komutanlığı ve Kara Kuvvetleri Komutanlığı belirlenmiştir. Satıcı kurum olarak ise; Satıcı1, Satıcı2 ve Satıcı3 kurumları alınmıştır. Hava Kuvvetleri Komutanlığı ile satıcı kurumlar arasındaki kurumlar arası ticaret modellendiğinde, Kara Kuvvetleri Komutanlığı ile satıcı kurumlar arasındaki kurumlar arası ticaret modeli de kolaylıkla geliştirilebilir. Bu nedenle, tasarım detayları sadece Hava Kuvvetleri Komutanlığı için verilmiştir.

Şekil 7.1'de kurumlar arası elektronik ticaret modelimizin ağ yapısı verilmiştir. Buna göre, kurumlar arasındaki mesaj alışverişi, internet üzerinden ve XML dokümanları kullanılarak gerçekleştirilmektedir.

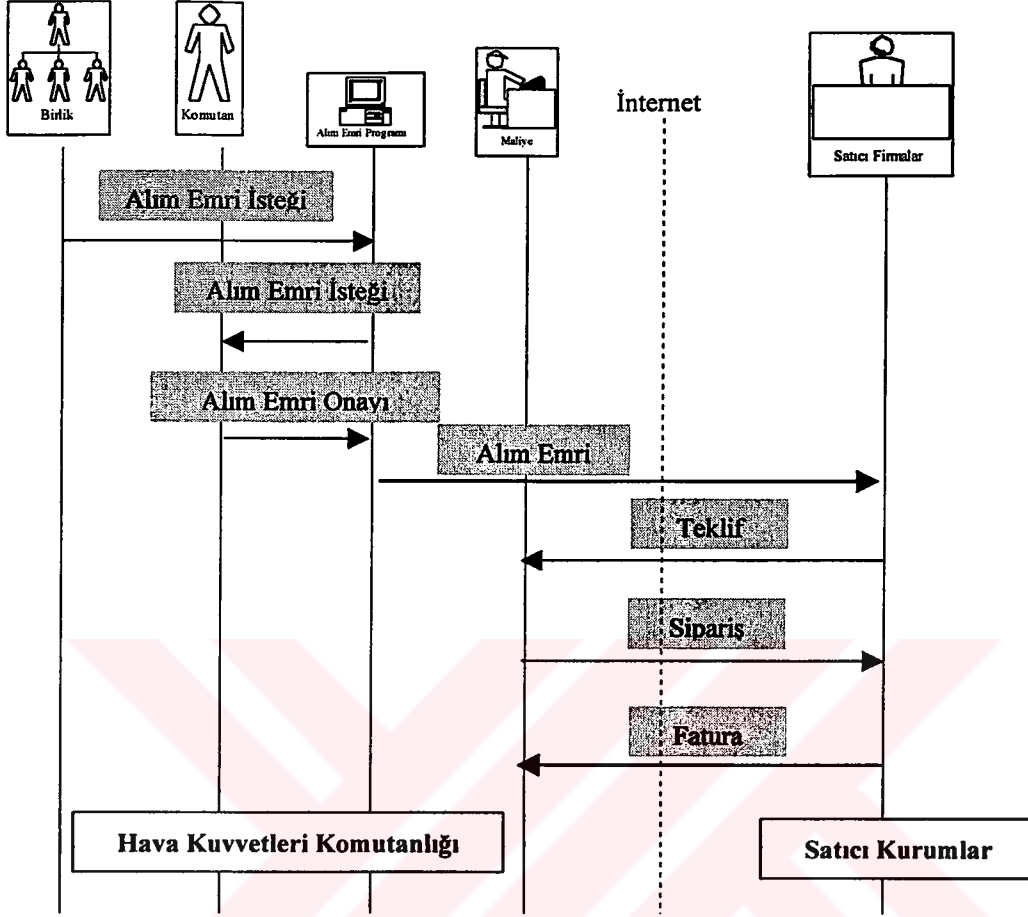
Bir kurumlar arası ticaret modelinde, çok çeşitli kurumlar rol almakta ve kurum sayısı arttıkça model karmaşıklaşmaktadır. Bu nedenle oluşturacağımız modelde, alıcı ve satıcı kurumlar arasında kurumlar arası elektronik ticaret gerçekleştirilmiştir. Bu sistemde, alıcı kurumun yapabileceği işlemler; alım emri isteğini oluşturmak, alım emri isteğini alım emrine çevirmek, oluşturulan alım emrini satıcı kurumlara göndermek; satıcı kurumlardan gelen teklifleri kaydetmek, bu teklifler arasında en ekonomik olanı seçen alım kararına göre uygun satıcı kuruma sipariş vermek, siparişi kazanan kuruma göndermektir. Satıcı kurumun yapabileceği işlemler ise; teklif oluşturmak, oluşturulan teklifleri alıcı kurumlara göndermek, sipariş almak, fatura göndermektir. Hava Kuvvetlerinin satıcı kurumlardan mal alım senaryosu, aktörleri ile birlikte aşağıda açıklanmıştır.



Şekil 7.1 Ağ şeması

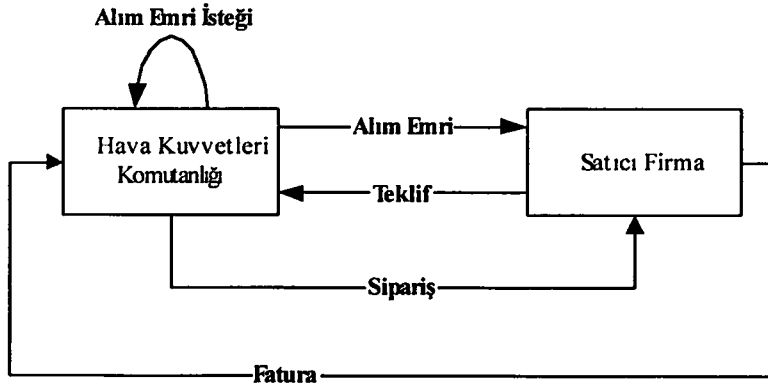
### 7.1 HKK.lığının Satıcı Kurumlardan Mal Alım Senaryosu

Hava Kuvvetleri Komutanlığındaki birlikler, alım emri isteklerini bilgisayara girmektedir. Bu dokümanlar, Hava Kuvvetleri Komutanlığı karargahındaki ana bilgisayara XML dokümanı olarak gönderilmektedir. Ana bilgisayardaki veritabanına, **alım emri istekleri** kaydedilir. Daha sonra alım emri istekleri komutana sunulmaktadır. Komutanın onayladığı alım istekleri **alım emri** olarak adlandırılır. Satıcı kurumlardan, oluşturulan alım emri gönderilerek, teklif istenir. Satıcı kurumlar, alım emrindeki malların her birine ne kadar fiyat verdiğini hem kendi veritabanına hem de gelen alım emrine kaydeder. Satıcı kurum oluşturduğu teklifi, Hava Kuvvetleri Komutanlığına göndermektedir. Yeterli sayıda satıcı kurumdan teklif alındıktan sonra, alım kararına göre kazanan kurum belirlenir. Modelimizde alım kararı, en ekonomik teklifin seçilmesine dayanmaktadır. Kazanan kurum, bu alım kararına göre, Maliye tarafından belirlendikten sonra, alım emrini **faturaya** çevirir. Hava Kuvvetleri Komutanlığının eklemesi gereken bilgiler faturaya kaydedilir ve satıcı kuruma sipariş verilir. Satıcı kurum, aldığı siparişi veritabanına kaydeder ve doldurması gereken bilgileri faturaya yazdıktan sonra, faturayı Hava Kuvvetleri Komutanlığına gönderir. Mal alım senaryosundaki dosya akışı, Şekil 7.2'de verilmiştir.



Şekil 7.2 HKKlığı-Satıcı Kurum arasındaki dosya akış şeması

Bu mal alım senaryosuna göre, kurumlar arası mesajlaşma, Şekil 7.3'te gösterilmiştir.



Şekil 7.3 Kurumlar arası mesajlaşma

Mesajlaşma, Şekil 7.3'teki gibi tasarlandıktan sonra, her aşamadaki mesajın hangi XML dosyası ile belirtileceği ve bu XML dosyasında hangi bilgilerin olması gerektiği tasarlanır. Bu

alım emri ve teklif aşamalarındaki bilgiler, alım emri XML dosyasında, sipariş ve fatura aşamalarındaki bilgiler, fatura XML dosyasında tutulacaktır.

## 7.2 Alım Emri İsteği XML Dosyası

Alım emri isteği XML dosyasında Hava Kuvvetleri Komutanlığına birliklerden gelen alım emri isteğinin tutulması tasarlanmaktadır. Bu dosyada başlık, çalışan bilgisi, mal ve toplam bilgileri bulunacaktır. Başlık bilgisi; alım emri isteğinin verildiği tarih, alım emri numarası ve alım emrini veren kurumun numarası bilgilerinden oluşmaktadır. Çalışan bilgisi; çalışanın soyadı, adı ve numarası bilgilerinden oluşmaktadır. Mal bilgisi; malın tanımı, universal kodu, ölçü birimi, kaç tane alınmasının istendiği ve satır numarası bilgilerinden oluşmaktadır. Satır numarası bilgisi ile çalışanın seçtiği sırada, alımı istenen malların alım emrinde gösterimi sağlanmaktadır. Toplam bilgisi ile toplam kaç kalem malın alımının istendiği bilgisi tutulmaktadır. Buna göre alım emri isteği XML dosyası, Şekil 7.4'teki bilgileri tutacaktır.

```

AlEmIst
  Baslik
    Tarih: date
    AlEmNo: integer
    KurumNo: integer
  CalisanBilgisi
    Soyad: string(30)
    Ad: string(30)
    CalisanNo: integer
  Mal +
    Tanim: string(30)
    UniversalKod: integer
    OlcuBirimi: string(2)
    Adet: integer
    SatirNo: integer
  Toplam
    KacKalemMal: integer

```

Şekil 7.4 Alım emri isteği XML dosyasında tutulan veriler

Burada kurum numaralarının ve mal bilgilerinin, tüm alıcı-satıcı kurumların veritabanlarında aynı olduğu varsayılmaktadır. Alım emri isteğinde, en az bir adet mal tanımı vardır. Bu bildirim “+” işareti ile gösterilmiştir. Bu şekilde tanımlanan alım emri isteği dokümanı için BizTalk Editör, Şekil 7.5'te verilen DTD'yi yaratmaktadır.



```

<?xml version="1.0" ?>
<!--
    Microsoft BizTalk Editor - Microsoft Corporation
    Microsoft Corporation, 2000 (http://www.microsoft.com)
-->
<Schema name="AIEmIst" b: BizTalkServerEditorTool_Version="1.0"
  b:root_reference="AIEmIst" b:standard="XML" xmlns="urn:schemas-microsoft-com:xml-data" xmlns:b="urn:schemas-microsoft-com: BizTalkServer"
  xsl:base="urn:schemas-microsoft-com:datatypes">
  <b: SelectionFields />
  <ElementType name="Toplam" content="empty" model="closed">
    <b: RecordInfo />
    <AttributeType name="KacKalemMal" datatype="int">
      <b: FieldInfo />
    </AttributeType>
    <attribute type="KacKalemMal" required="no" />
  </ElementType>
  <ElementType name="Mal" content="empty" model="closed">
    <b: RecordInfo />
    <AttributeType name="UniversalKod" datatype="int">
      <b: FieldInfo />
    </AttributeType>
    <AttributeType name="Tanim" datatype="string" d:maxLength="30"
      d:minLength="1">
      <b: FieldInfo />
    </AttributeType>
    <AttributeType name="SatirNo" datatype="int">
      <b: FieldInfo />
    </AttributeType>
    <AttributeType name="OlcuBirimi" datatype="string" d:maxLength="2"
      d:minLength="1">
      <b: FieldInfo />
    </AttributeType>
    <AttributeType name="Adet" datatype="int">
      <b: FieldInfo />
    </AttributeType>
    <attribute type="Tanim" required="no" />
    <attribute type="UniversalKod" required="no" />
    <attribute type="OlcuBirimi" required="no" />
    <attribute type="Adet" required="no" />
    <attribute type="SatirNo" required="no" />
  </ElementType>
  <ElementType name="CalisanBilgisi" content="empty" model="closed">
    <b: RecordInfo />
    <AttributeType name="Soyad" datatype="string" d:maxLength="30"
      d:minLength="1">
      <b: FieldInfo />
    </AttributeType>
    <AttributeType name="CalisanNo" datatype="int">
      <b: FieldInfo />
    </AttributeType>
    <AttributeType name="Ad" datatype="string" d:maxLength="30"
      d:minLength="1">
      <b: FieldInfo />
    </AttributeType>
    <attribute type="Soyad" required="no" />
    <attribute type="Ad" required="no" />
    <attribute type="CalisanNo" required="no" />
  </ElementType>

```

```

</ElementType>
<ElementType name="Baslik" content="empty" model="closed">
  <b:RecordInfo />
  <AttributeType name="Tarih" d:type="date">
    <b:FieldInfo />
  </AttributeType>
  <AttributeType name="KurumNo" d:type="int">
    <b:FieldInfo />
  </AttributeType>
  <AttributeType name="AIEmNo" d:type="int">
    <b:FieldInfo />
  </AttributeType>
  <attribute type="Tarih" required="no" />
  <attribute type="AIEmNo" required="no" />
  <attribute type="KurumNo" required="no" />
</ElementType>
<ElementType name="AIEmIst" content="eltOnly" model="open">
  <b:RecordInfo />
  <element type="Baslik" maxOccurs="1" minOccurs="1" />
  <element type="CalisanBilgisi" maxOccurs="1" minOccurs="1" />
  <element type="Mal" maxOccurs="*" minOccurs="1" />
  <element type="Toplam" maxOccurs="1" minOccurs="1" />
</ElementType>
</Schema>

```

Şekil 7.5 Alım emri isteği dokümanının DTD'si

### 7.3 Alım Emri XML Dosyası

Alım emri XML dosyası ile Hava Kuvvetleri Komutanlığı ile satıcı kurumlar arasında teklif aşamasında kullanılan bilgilerin tutulması tasarlanmaktadır. Bir alım emri XML dosyası, Şekil 7.6'daki bilgileri tutacaktır.

```

AIEm
  Baslik
    Tarih: date
    AIEmNo: integer
    KurumNo: integer
  CalisanBilgisi
    Soyad: string(30)
    Ad: string(30)
    CalisanNo: integer
  Mal+
    Tanım: string(30)
    UniversalKod: integer
    OlcuBirimi: string(2)
    Adet: integer
    Fiyat: real
    HesaplanmisFiyat: real
    SatirNo: integer
  Toplam
    KacKalemMal: integer
    AIEmToplami: integer

```

Şekil 7.6 Alım emri XML dosyasında tutulan veriler

Görüldüğü gibi bir alım emri, alım emri isteği ile tutulan bilgileri içermektedir. Ayrıca; mal bilgisinde fiyat, hesaplanmış fiyat ve toplam bilgisinde de alım emri toplamı bilgilerini de tutmaktadır. Hesaplanmış fiyat ile adet ve fiyat bilgisinin çarpımı tutulmaktadır.

Hava Kuvvetleri Komutanlığına teklif veren her kurum, her mal için verdiği fiyat ile hesaplanmış fiyatı ve alım emri toplamını XML dosyasına eklemektedir.

Alım emri dokümanının DTD'si Ek 3'te verilmiştir.

#### **7.4 Fatura XML Dosyası**

Fatura XML dosyasında Hava Kuvvetleri Komutanlığı ile satıcı kurumlar arasında sipariş ve fatura aşamasında kullanılan bilgilerin tutulması tasarlanmaktadır. Fatura dosyası başlık, satıcı, alıcı, fatura satış şartları, mal, toplam, fatura adresi, gönderme adresi bilgilerinden oluşmaktadır. Başlık; fatura tarihi, fatura numarası (modelimizde satıcı kurumun verdiği teklif numarası fatura numarası yerine kullanılmıştır), alıcı kurumun alım emri numarası, alım emri tarihi, satıcı kurumun malı gönderme tarihi bilgilerinden oluşmaktadır. Satıcı bilgisi, adres ve kontakt bilgilerinden oluşmaktadır. Alıcı bilgisi, adres ve kontakt bilgisinden oluşmaktadır. Kontakt bilgisi ile ilgili kurumda iletişime geçeceğimiz çalışanın adı ve numarası tutulmaktadır. Fatura satış şartları bilgisi ile satıcı kurumun toplam ne kadar indirim yaptığı bilgisi tutulmaktadır. Mal bilgisi, alım emrinde olduğu gibi tutulmaktadır. Toplam bilgisi ile toplam kaç kalem mal alındığı, alım emri toplamı, toplam indirim miktarı, vergi miktarı, fatura toplamı tutulmaktadır. Fatura adresi ile fatura edilen adres, gönderme adresi ile malların gönderildiği yer tutulmaktadır. Buna göre fatura XML dosyası ile tutulacak bilgiler, Şekil 7.7'de verilmiştir. Fatura dosyasının DTD'si Ek 3'te verilmiştir.

**Fatura****Baslik**

Tarih: date  
 Numara: integer  
 AEmNo: integer  
 AEmTarih: date  
 GondermeTarihi: date

**Satici****Adres**

Ad: string(60)  
 Adres1: string(55)  
 Adres2: string(55)  
 Il: string(30)  
 PostaKodu: string(15)

**KontaktBilgisi**

Ad: string(60)  
 CalisanNo: integer

**Alici****Adres**

Ad: string(60)  
 Adres1: string(55)  
 Adres2: string(55)  
 Il: string(30)  
 PostaKodu: string(15)

**KontaktBilgisi**

Ad: string(60)  
 CalisanNo: integer

**FaturaSatisSartlari**

IndirimOrani: real

**Mal+**

Tanim: string(30)  
 UniversalKod: integer  
 OlcuBirimi: string(2)  
 Adet: integer  
 Fiyat: real  
 HesaplanmisFiyat: real  
 SatirNo: integer

**Toplam**

KacKalemMal: integer  
 AEmToplami: real  
 IndirimMiktari: real  
 VergiMiktari: real  
 FaturaToplami: real

**FaturaAdresi****Adres**

Ad: string(60)  
 Adres1: string(55)  
 Adres2: string(55)  
 Il: string(30)  
 PostaKodu: string(15)

**GondermeAdresi****Adres**

Ad: string(60)  
 Adres1: string(55)  
 Adres2: string(55)  
 Il: string(30)  
 PostaKodu: string(15)

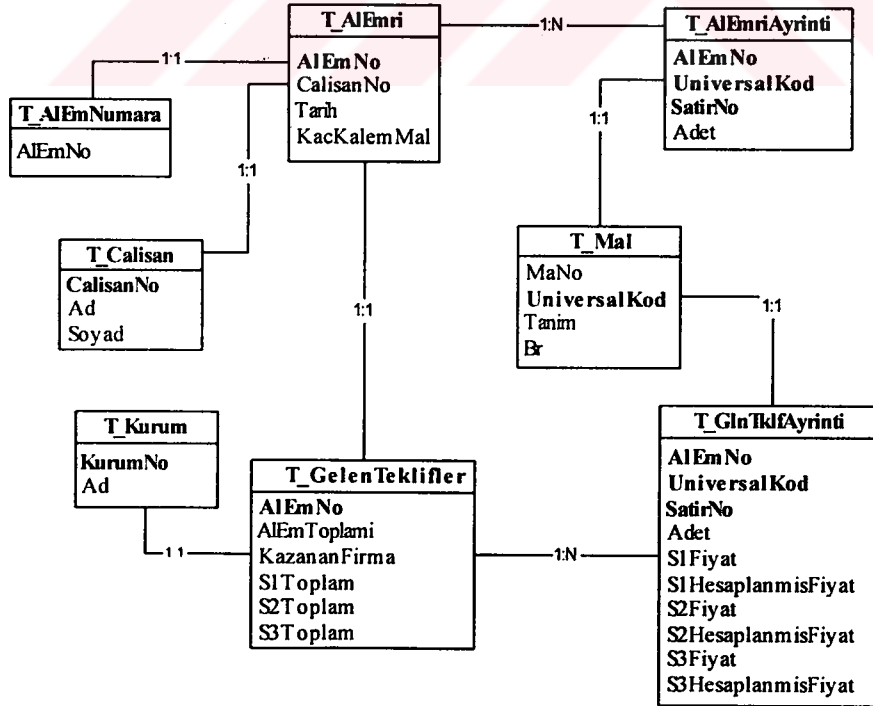
Şekil 7.7 Fatura XML dosyasında tutulan veriler

## 7.5 HKK Veritabanı

XML dosyaları bu şekilde tasarlandıktan sonra, bu bilgilerin veritabanında nasıl saklanacağı belirlenir. Ek-1'de HKK veritabanı verilmiştir. Şekil 7.8'de HKK veritabanının Varlık-İlişki şeması verilmiştir. Koyu yazılan alanlar, o tablonun anahtarını göstermektedir.

HKK veritabanında bir alım emri numarası bir defa kullanılacaktır. Bunun için verilen en son alım emri numarası T\_AIEmNumara tablosunda AIEmNo alanında tutulacaktır. Bu tabloda sadece bir kayıt bulunmaktadır. Alım emri isteği yaratılırken buradan numara alınacak ve AIEmNo 1 arttırılarak tabloda değiştirilecektir.

Her alım emri isteğini kimin gerçekleştirdiği, alım emrinde kayıtlı olacaktır. Bu da CalisanNo bilgisinin T\_AIEmri tablosunda tutulması ile sağlanacaktır. T\_AIEmri tablosu ile alım emrinde bulunan ve sadece bir defa tekrar eden veriler tutulacaktır. T\_AIEmriAyrinti tablosu ile alınması istenen malların bilgileri tek tek tutulacaktır. T\_AIEmriAyrinti tablosunda anahtar; AIEmNo, UniversalKod ve SatirNo'dur. SatirNo sayesinde aynı mal birden fazla olarak bir alım emri isteğinde bulunabilecektir.



Şekil 7.8 HKK veritabanı Varlık-İlişki şeması

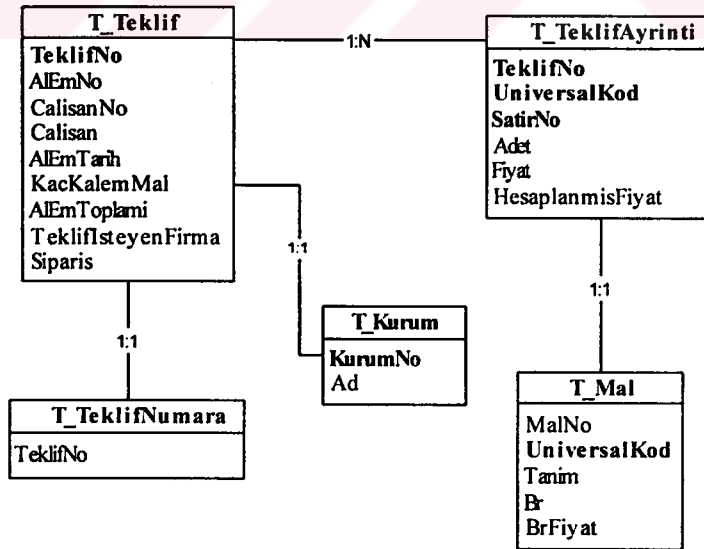
Satıcı kurumlardan gelen teklifler T\_GelenTeklifler tablosunda tutulacaktır. Her alım emri için bir gelen teklif, veritabanında kayıtlı olacaktır. T\_GelenTeklifler tablosunda Satıcı1, Satıcı2 ve Satıcı3 kurumlarının verdiği teklif toplamları kaydedilecektir. Ayrıca bu tabloda kazanan kurumun kodu ve verdiği teklif tutulacaktır. Böylece hızlı bir şekilde kazanan kurum sorgulanabilecektir. T\_GlnTklfAyrınti tablosu ile her kurumun her mal için verdiği teklif tutulacaktır.

T\_Mal tablosunda anahtar, universalKod'tur. Fakat, HKKlığı kendi içinde de bu mala kod vermiş olabilir. Bu nedenle MalNo bilgisi tutulmaktadır.

T\_Kurum tablosu, gelen mesajların hangi kurum tarafından gönderildiğini anlamak ve yaratılan mesajın hangi kuruma gönderileceğini belirlemek için tutulacaktır.

## 7.6 Satıcı1 Veritabanı

Ek-2'de Satıcı1 veritabanı verilmiştir. Şekil 7.9'da Satıcı1 veritabanının Varlık-İlişki şeması verilmiştir. Koyu yazılan alanlar, o tablonun anahtarını göstermektedir.



Şekil 7.9 Satıcı1 veritabanı Varlık-İlişki şeması

Satıcı1 veritabanında bir teklif numarası bir defa kullanılacaktır. Bunun için verilen en son teklif numarası T\_TeklifNumara tablosunda TeklifNo alanında tutulacaktır. Bu tabloda sadece

bir kayıt bulunmaktadır. Alıcı kurumlara teklif verilirken buradan numara alınacak ve TeklifNo 1 artırılarak tabloda deęiştirilecektir.

Her sipariřin hangi alıřan tarafından alındığı faturada kayıtlı olacaktır. Bu da CalisanNo bilgisinin T\_Teklif tablosunda tutulması ile saęlanacaktır. T\_Teklif tablosu ile teklifte bulunan ve sadece bir defa tekrar eden veriler tutulacaktır. T\_TeklifAyrıntı tablosu ile istenen mal bilgileri ve Satıcı1 kurumun verdięi fiyatlar tek tek tutulacaktır.

T\_Kurum tablosu, gelen mesajların hangi kurum tarafından gönderildiğini anlamak ve yaratılan mesajın hangi kuruma gönderileceğini belirlemek için tutulacaktır.

T\_Mal tablosunda anahtar, UniversalKod'tur. Fakat, Satıcı1 kendi içinde de bu mala kod vermiş olabilir. Bu nedenle MalNo bilgisi tutulmaktadır.

Satıcı2 ve Satıcı3 kurumlarının veritabanı da Satıcı1 veritabanının tasarlandığı gibi tasarlanacaktır.

## **7.7 Kurumlar Arasındaki Mesajlaşma**

Hava Kuvvetleri ile satıcı kurumlar arasındaki mesajlaşmanın saęlanabilmesi için; dokümanların, orgnizasyonların, portların ve yazılım paralarının tanımlanması gerekmektedir.

### **7.7.1 Dokümanlar**

Tasarladığımız XML dosyalarını, mesajlaşma için doküman olarak ařağıdaki şekilde tanımlarız.

<b>T_AlımEmriIsteęi</b>	<b>T_AIEmIst.xml</b>
<b>T_AlımEmri</b>	<b>T_AIEm.xml</b>
<b>T_Fatura</b>	<b>T_Fatura.xml</b>

### 7.7.2 Organizasyonlar

Geliştirdiğimiz modelde 5 kurum bulunmaktadır: HKK, KKK, Satıcı1, Satıcı2, Satıcı3. Bu kurumlar, Alıcı ve Satıcı üzerinde tek tek organizasyon olarak tanımlanmalıdır. Alıcı kurum olarak Hava Kuvvetleri Komutanlığı kurumunu tanımlarsak; HKK organizasyonunu hem satıcı hem de alıcı kurumlar üzerinde tanımlamamız gerekmektedir:

T\_HKK on Buyer  
 T\_HKK on Seller(Saticı1)  
 T\_HKK on Seller(Saticı2)  
 T\_HKK on Seller(Saticı3)

Saticı1 kurumunu modelimizde satıcı kurum olarak tanımlarsak, aşağıdaki organizasyonları sistemimize dahil ederiz:

T\_Saticı1 on Buyer(HKK)  
 T\_Saticı1 on Seller

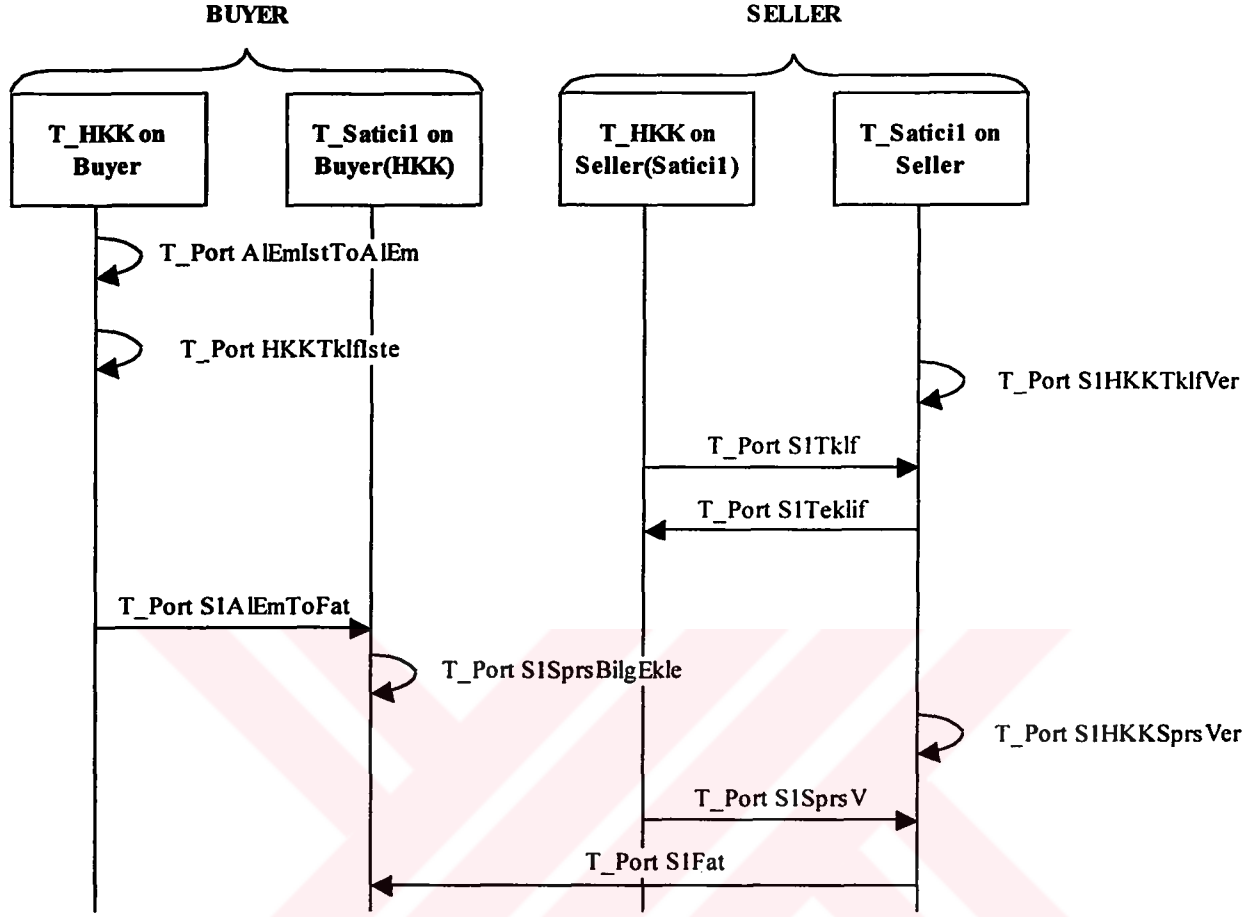
Bir mesajın her zaman için kaynağı ve de gideceği adresi bulunur. Örneğin HKK Satıcı1'e mesaj göndermek isterse; bu mesajın yaratıcısı kendisi, "T\_HKK on Buyer", gideceği yer de kendi üzerindeki Saticı1, "T\_Saticı1 on Buyer(HKK)" olur. Daha sonra gönderilen mesaj XML şekliyle "T\_Saticı1 on Seller" a internet üzerinden gönderilir. Tüm organizasyon tanımları Ek 9'da verilmiştir.

Modelimize göre alıcı (HKK) ve satıcı (Saticı1) kurum arasında mal alımının gerçekleşebilmesi için Şekil 7.10'da gösterilen organizasyonlar arasında belirtilen portların tanımlanması gerekmektedir. Şekildeki en üst sıradaki kutuların içinde, organizasyon adları verilmiştir.

### 7.7.3 Portlar

Port tanımları ile mesajın hangi organizasyondan hangi organizasyona gönderildiği, kullanılan kanal, çalışacak uygulama tanımlanmaktadır. Hava Kuvvetleri Komutanlığı ile Satıcı1 kurumu arasında tanımlı portlar, Hava Kuvvetleri ile Satıcı2 ve Satıcı3 kurumları arasında da tanımlanmıştır. Şekil 7.10'da verilen port tanımları aşağıda açıklanmaktadır. Tanımlı tüm portlar Ek 11'de verilmiştir.





Şekil 7.10 Organizasyonlar arasında mesajlaşma ve kullanılan portlar

### 7.7.3.1 T\_Port AEmIstToAEm

“T\_HKK on Buyer” organizasyonu T\_AlEmIstegi dokümanını, T\_AlEmri dokümanına çevirir. Yeni oluşturulan T\_AlEmri dokümanı, “file:///d:\inetpub\wwwroot\tez\AEm\alEm%tracking\_id%.xml” olarak kaydedilir. “%tracking\_id%”, tek numara elde etmek için kullanılmaktadır. Böylece “AEm” harfleri ile başlayan ve tek numara ile devam eden bir dosya adı elde edilir ve dosya belirtilen dizinde yaratılır.

### 7.7.3.2 T\_Port HKKTklfIste

“T\_HKK on Buyer” organizasyonu T\_AlিমEmri dokümanını, “HKKTklfIste Sprs” uygulama entegrasyon yazılım parçası ile Satıcı1, Satıcı2, Satıcı3 kurumlarının teklif dizinlerine, yazar. Satıcı kurumlar, teklif dizinlerine gelen dokümanlar ile bu dizinlerde tanımlı dosya alım fonksiyonları ile teklif iş akışlarını etkinleştirirler.

### 7.7.3.3 T\_Port S1HKKTklfVer

“T\_Saticı1 on Seller” organizasyonu teklif dizinine gelen T\_AlımEmri dokümanı ile teklif iş akışını, “S1TeklifHKKQ TklfSchedule” uygulama entegrasyon yazılım parçası ile etkinleştirir. Bu yazılım parçası, teklif iş akışının çalışması için T\_AlımEmri dokümanını, S1Teklif kuyruğuna yazar ve Satıcı1’in teklif iş akışını çalıştırır.

### 7.7.3.4 T\_Port S1Tklf

“T\_HKK on Seller (Saticı1)” organizasyonu T\_AlımEmri dokümanını, “T\_Saticı1 on Seller” organizasyonuna gönderir. T\_AlımEmri dokümanı “S1TeklifYazAIC S1Teklif” uygulama entegrasyon parçasına verilir. Satıcı1 veritabanına T\_AlımEmri dokümanı, teklif olarak kaydedilir ve gelen T\_AlımEmri dokümanı üzerine her mal için Satıcı1’in verdiği fiyat yazılır.

### 7.7.3.5 T\_Port S1Teklif

“T\_Saticı1 on Seller” organizasyonu T\_AlımEmri dokümanını, “T\_HKK on Seller(Saticı1)” organizasyonuna gönderir. T\_AlımEmri dokümanı “http://study/HS1FiyatQ/saticı1MSMQ.asp” verilir. HKKS1 kuyruğuna T\_AlımEmri dokümanı kaydedilerek Satıcı1 den HKKlığına teklif gönderilmiş olur.

#### 7.7.3.6 T\_Port S1AlEmToFat

Sipariş verilecek kurum belirlendikten sonra, “T\_HKK on Buyer” organizasyonu T\_AlEmri dokümanını, T\_Fatura dokümanına çevirir. T\_Fatura dokümanı “file://d:\Inetpub\wwwroot\tez\SiparisS1\ilkSprs%tracking\_id%.xml” olarak kaydedilir ve T\_Fatura, “T\_Satici1 on Buyer(HKK)” organizasyonuna gönderilir.

#### 7.7.3.7 T\_Port S1SprsBilgEkle

“T\_Satici1 on Buyer(HKK)” organizasyonu, HKKlığının faturada doldurması gereken bilgileri doldurmak için T\_Fatura dokümanını, “HKKSprS1 SprsS1” uygulama entegrasyon yazılım parçasına verir.

#### 7.7.3.8 T\_Port S1HKKSprVer

“T\_Satici1 on Seller” organizasyonu, T\_Fatura dokümanını “S1SprsHKKQ SprsSchedule” uygulama entegrasyon yazılım parçası ile sipariş vermek için, Satici1’in sipariş dizinine yazar. Bu dizinde Satici1’in tanımladığı dosya alım fonksiyonu ile sipariş iş akışı etkinleştirilir.

#### 7.7.3.9 T\_Port S1SprsV

“T\_HKK on Seller(Satici1)” organizasyonu T\_Fatura dokümanını, “T\_Satici1 on Seller” organizasyonuna gönderir. T\_Fatura dokümanı “S1SprsHKKAIC Satici1Sprs” uygulama entegrasyon parçasına verilir. Satici1 veritabanına, T\_Fatura dokümanındaki alım emri numarasına göre, sipariş verildi olarak T\_Fatura dokümanı kaydedilir.

### 7.7.3.10 T\_Port S1Fat

“T\_Satici1 on Seller” organizasyonuna gelen T\_Fatura dokümanı, “T\_Satici1 on Buyer(HKK)” organizasyonuna gönderilir. Gönderilen fatura, HKKlığının “file:///d:/inetpub/wwwroot/tez/fatura/fat%tracking\_id%.xml” dizinine kaydedilir.

## 7.7.4 COM parçaları

Geliştirilen elektronik ticaret sisteminde, Hava Kuvvetleri Komutanlığı alıcı kurumunun ve Saticı1 satıcı kurumunun COM parçaları ve açıklamaları aşağıda verilmiştir.

### 7.7.4.1 Basla.Tezim

Bu kod ile HKK.xml veri dosyası HKKAlEmIst kuyruğuna atılır. Satıcı kurumlardan teklif isteyen, bu teklifleri kaydeden, bu teklifler arasında teklif toplamı en küçüğü belirleyen ve belirlenen bu kuruma sipariş veren HKK.skx iş akışı çalıştırılır.

### 7.7.4.2 HKKSpsS1.SprsS1

Bu kod ile Saticı1 kurumuna sipariş verme işlemleri başlatılmaktadır. Fatura üzerinde HKKlığının doldurması gereken fatura adresi, Hava Kuvvetleri Komutanlığı; gönderme adresi ise Hava Harp Okulu Komutanlığı olarak doldurulur. Siparişi kazanan kurum, Saticı1 kurumu olarak HKK veritabanına kaydedilir. HKKlığının SiparisS1, Saticı1 kurumunun Siparis dizinine oluşturulan sipariş kaydedilir.

### 7.7.4.3 HKKSpsS2.SprsS2

Bu kod ile Saticı2 kurumuna sipariş verme işlemleri başlatılmaktadır. Fatura üzerinde HKKlığının doldurması gereken fatura adresi, Hava Kuvvetleri Komutanlığı; gönderme adresi ise Hava Harp Okulu Komutanlığı olarak doldurulur. Siparişi kazanan kurum, Saticı2

kurumu olarak HKK veritabanına kaydedilir. HKK.lığının SiparisS2, Satıcı2 kurumunun Siparis dizinine oluşturulan sipariş kaydedilir.

#### **7.7.4.4 HKKSpsS3.SprsS3**

Bu kod ile Satıcı3 kurumuna sipariş verme işlemleri başlatılmaktadır. Fatura üzerinde HKK.lığının doldurması gereken fatura adresi, Hava Kuvvetleri Komutanlığı; gönderme adresi ise Hava Harp Okulu Komutanlığı olarak doldurulur. Siparişi kazanan kurum, Satıcı3 kurumu olarak HKK veritabanına kaydedilir. HKK.lığının SiparisS3, Satıcı3 kurumunun Siparis dizinine oluşturulan sipariş kaydedilir.

#### **7.7.4.5 HKKTeklif.TeklifKaydet**

Bu kod ile HKK.lığına satıcı kurumların verdiği teklifler kaydedilmektedir.

#### **7.7.4.6 HKKTklflste.Sprs**

Bu kod ile satıcı kurumlardan teklif istenmektedir. Bunun için Satıcı1, Satıcı2 ve Satıcı3 kurumlarının Teklif dizinlerine alım emri, teklif\*.xml olarak kaydedilmektedir.

#### **7.7.4.7 S1SprsHKKAIC.Satici1Sprs**

Bu kod ile Satıcı1 kurumuna sipariş verilmektedir. Satıcı1 veritabanına, sipariş istendi olarak kaydedilmektedir. Daha sonra Satıcı1, faturada doldurması gereken; fatura tarihi, fatura numarası, siparişi verilen malları gönderme tarihi bilgilerini doldurmaktadır. Satıcı1, HKK.lığına %5 indirim uygulamaktadır ve bu tutarı hesaplayıp fatura indirim miktarı olarak faturaya kaydetmektedir. Ayrıca %10 vergi miktarını da faturaya eklemektedir. Daha sonra oluşturulan fatura dokümanı, HKK.lığının fatura dizinine kaydedilmesini sağlayan kanala verilmektedir.

#### 7.7.4.8 S1SprsHKKQ.SprsSchedule

Satıcı1 kurumuna gelen sipariş dokümanı, S1Siparis kuyruğuna atılmakta ve S1HKKsiparis.skx iş akışı çalıştırılmaktadır.

#### 7.7.4.9 S1TeklifHKKQ.TklfSchedule

Satıcı1 kurumuna gelen teklif dokümanı, S1Teklif kuyruğuna atılmakta ve S1HKKTeklif.skx iş akışı çalıştırılmaktadır.

#### 7.7.4.10 S1TeklifYazAIC.S1Teklif

Satıcı1 kurumu, verdiği teklifleri kendi veritabanına ve gelen dokümana kaydeder. Daha sonra doküman, alıcı kurumun Satıcı1'e ayırdığı teklif kuyruğuna dokümanı yazmayı sağlayan kanala verilir. Örneğin HKK.lığı için Satıcı1, HKKS1 kuyruğuna dokümanı yazmaktadır.

### 7.8 Parçaların BizTalk Orchestration Designer Aracı ile Bütünleştirilmesi

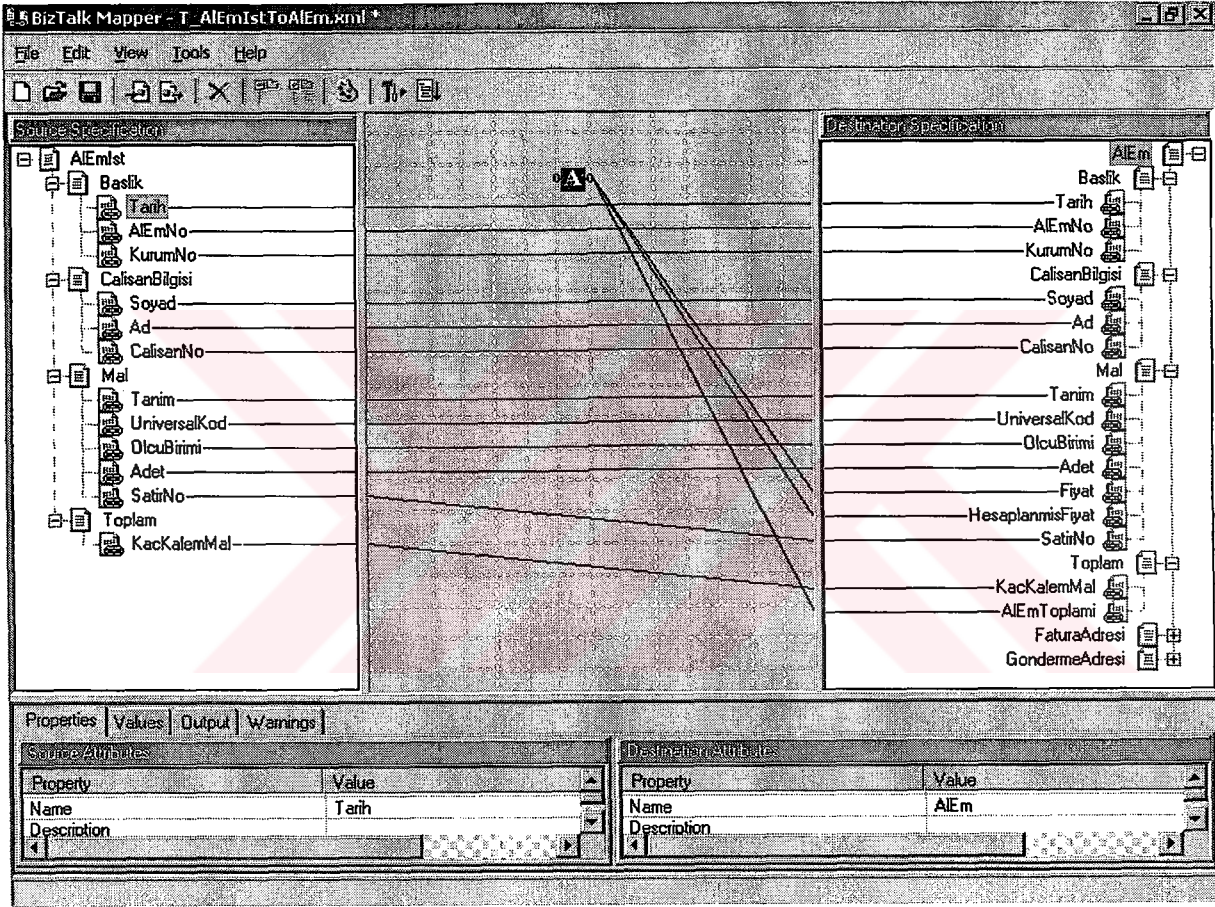
Buraya kadar tanımlanan tüm parçaların bir araya gelerek, bir B2B elektronik ticaret uygulaması oluşturması için BizTalk Orchestration Designer ile bir bütün haline getirilmesi gerekmektedir.

Hava Kuvvetleri Komutanlığı'nın gelen alım emri isteklerini, ASP yazılım parçası ile HAlEmIst kuyruğa atmaktadır. Kuyruğa gelen dokümanı, BizTalk Orchestration Designer ile tanımlanan iş akışı işleyecektir.

#### 7.8.1 HKKlığının mal alım iş akışı

Tanımlayacağımız HKK.skv dosyası, HKKlığına gelen dokümanı kuyruktan alacak, bunu gelen alım emri isteği XML dokümanı olarak dizine, Ek 8'de verilen HKKAlEmIstYaz.wsc

windows script yazılım parçası ile kaydedecektir. Daha sonra gelen alım emri isteğini alım emrine çevirecektir. Bu eyleme, alım emri isteği XML dokümanında ve alım emri XML dokümanında aynı isimle tanımlanmış alanların eşleştirilmesi ile gerçekleştirilmiştir. Alım emri XML dokümanında kalan diğer alanlara, boş bilgi atanarak XML dokümanında oluşması sağlanmıştır. Alım emri isteğinin alım emrine çevrimi BizTalk Mapper ile gerçekleştirilmiş ve alım emri isteğinin alım emrine çevrildiği eyleme, T\_AIEmIstToAIEm, Şekil 7.11'de verilmiştir.

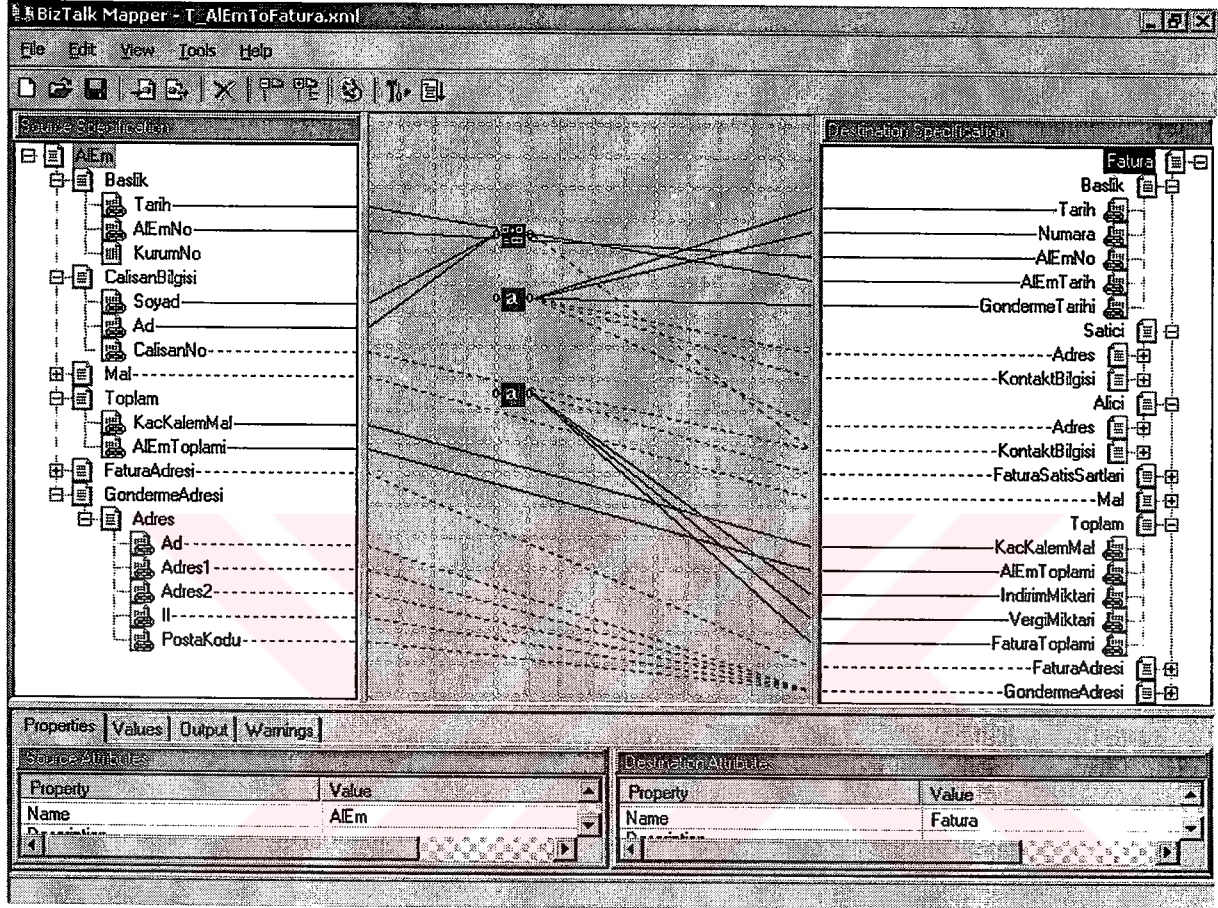


Şekil 7.11 Alım Emri İsteğinin Alım Emrine çevrimi

Çevrim gerçekleştikten sonra sırası ile Satıcı1, Satıcı2 ve Satıcı3 kurumlarından tekliflerin gelmesi beklenmektedir. Bu teklifler geldikten sonra, Satıcı1'in verdiği teklif S1Teklif dizinine, Satıcı2'nin verdiği teklif S2Teklif dizinine, Satıcı3'ün verdiği teklif S3Teklif dizinine, Ek 8'de verilen HKKTeklifYaz.wsc windows script yazılım parçası ile kaydedilecektir.

Hangi kurumun verdiği teklifteki alım emri toplamı küçükse o kurum, alım emrini faturaya çevirmektedir. BizTalk Mapper ile tanımlanan T\_AIEmToFatura eylemesi, Şekil 7.12'de

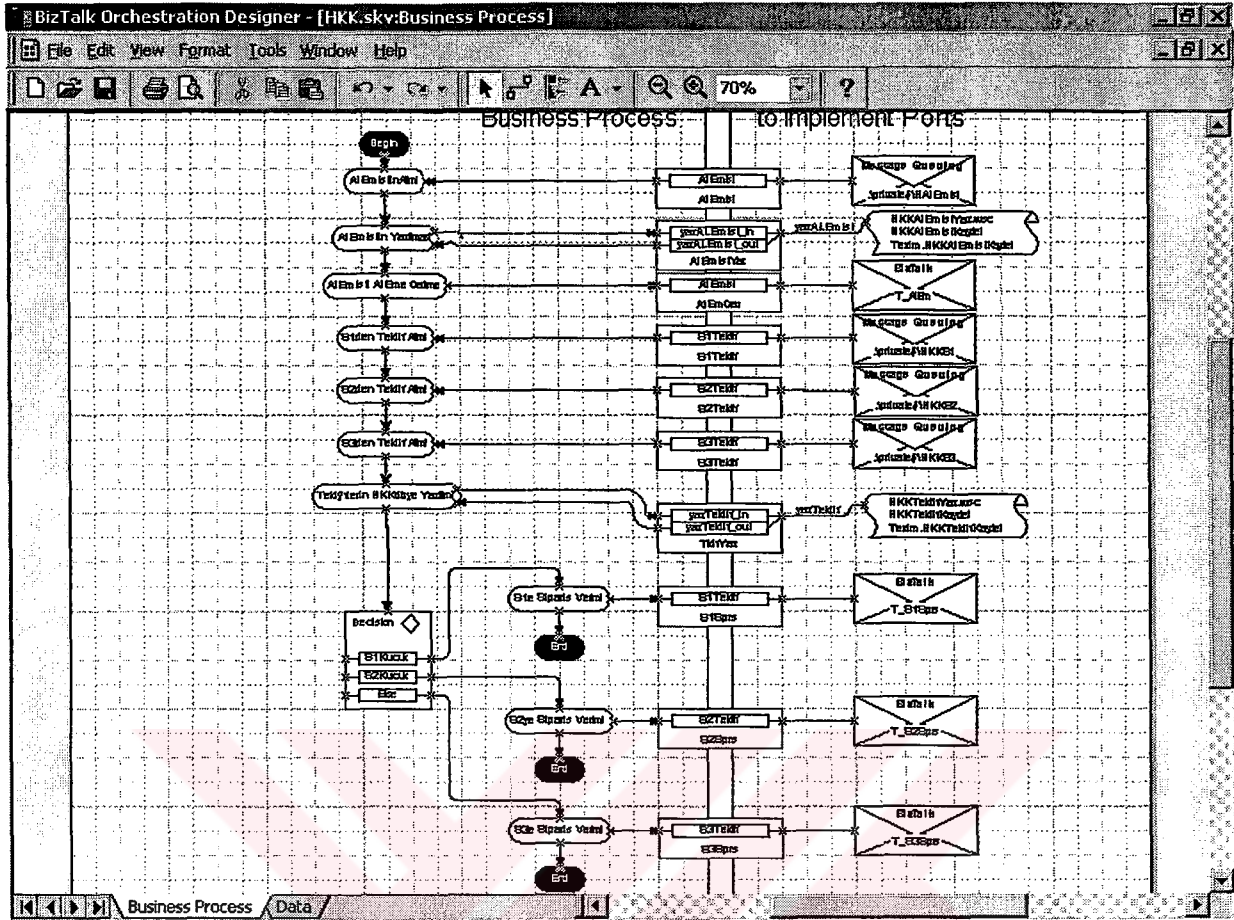
verilmiştir. Bu eşleme, alım emri XML dokümanında ve fatura XML dokümanında aynı isimle tanımlanmış alanların eşleştirilmesi ile gerçekleştirilmiştir. Fatura XML dokümanında kalan diğer alanlara boş bilgi atanarak, XML dokümanında oluşması sağlanmıştır.



Şekil 7.12 Alım Emrinin Faturaya çevrimi

Kazanan kurumun, alım emrini faturaya çevirmesi ile iş akışı sonlanır. BizTalk Orchestration Designer ile hazırlanan HKK.skv iş akışı, Şekil 7.13'te verilmiştir.

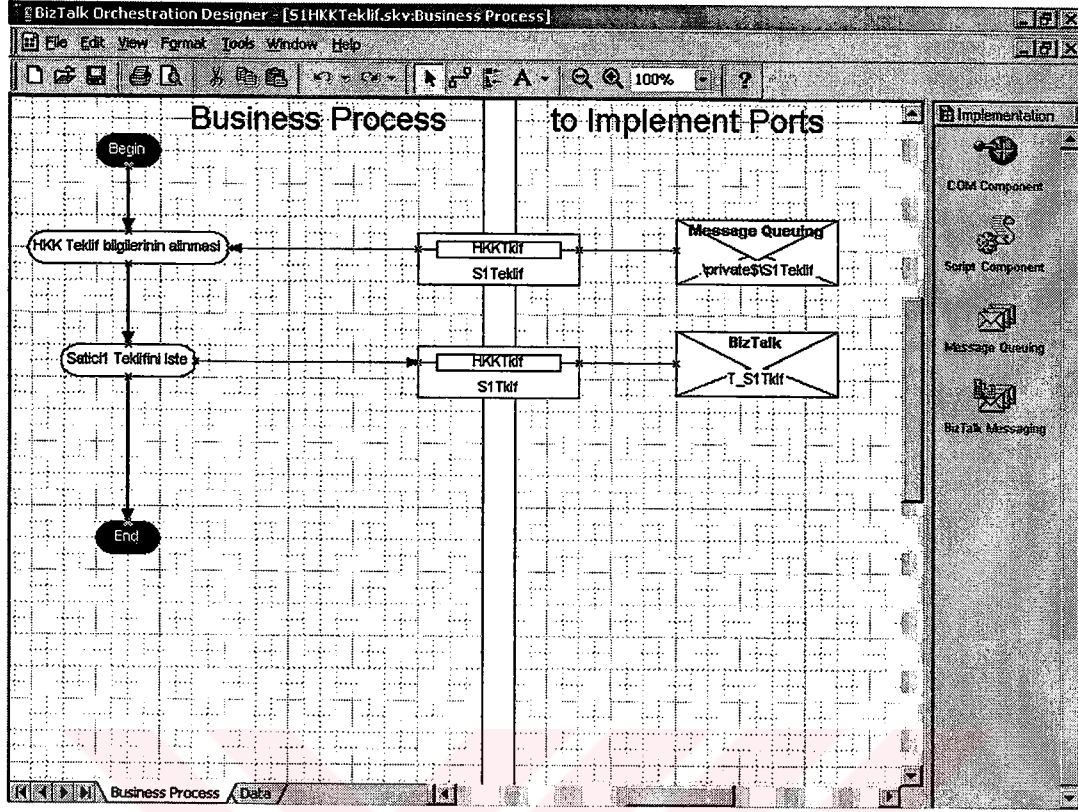




Şekil 7.13 HKK.skv iş akışı

## 7.8.2 Satıcı1'in teklif verme iş akışı

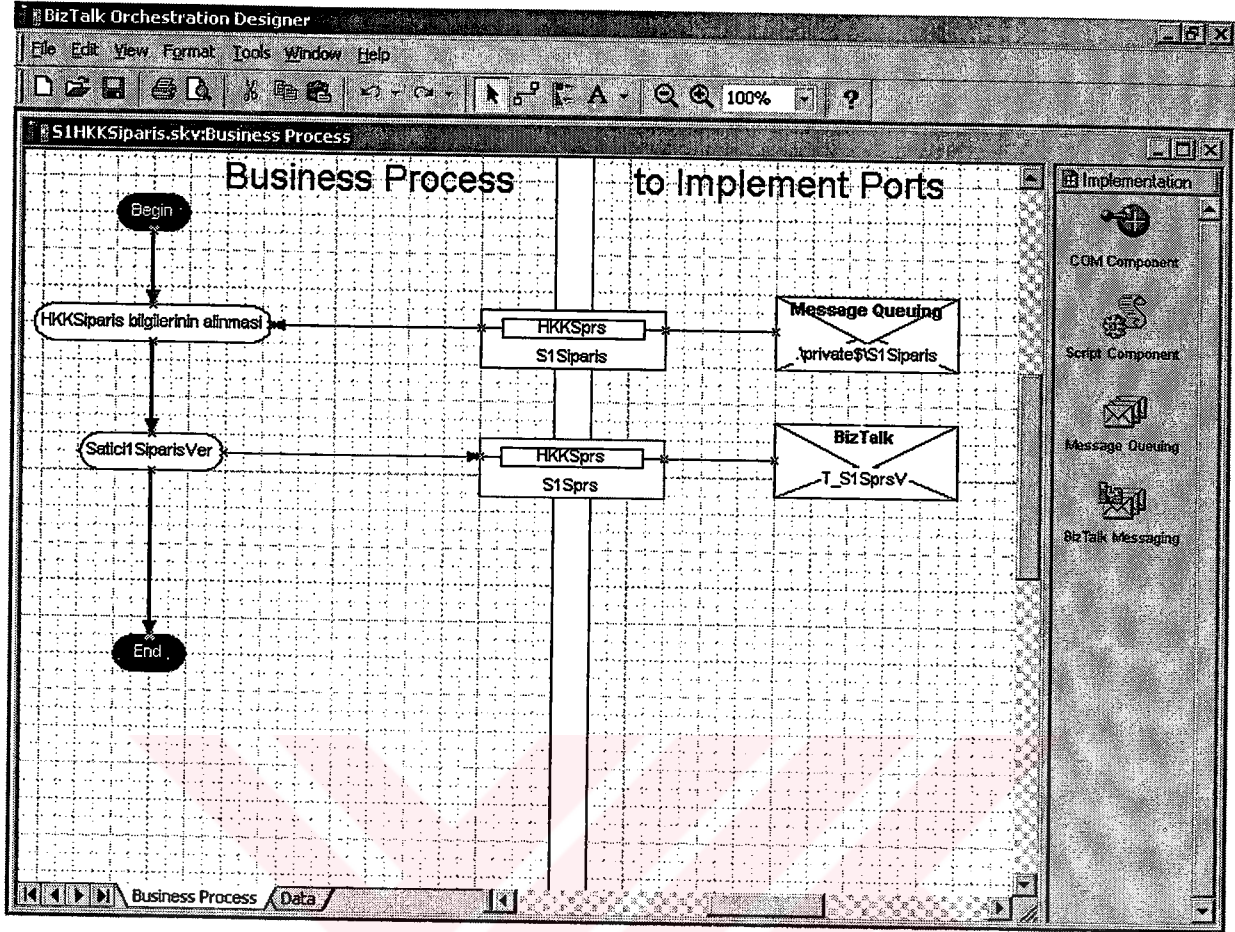
T\_Port S1HKKtklfVer portu ile Satıcı1 teklif iş akışı çalıştırılır. Satıcı1 verdiği teklifi veritabanına yazdıktan sonra, T\_Port S1Teklif portu ile HKK'daki S1Teklif kuyruğuna fatura XML dokümanını atar. Satıcı1 kurumunun teklif iş akışı Şekil 7.14'te verilmiştir. Buna göre, S1Teklif kuyruğundan XML dokümanı alınmakta ve T\_S1Tklf kanalı ile Satıcı1 kurumunun verdiği teklifler Satıcı1 veritabanında kaydedilmekte ve verilen teklif XML şeklinde Hava Kuvvetleri Komutanlığına gönderilmektedir.



Şekil 7.14 Satıcı1 kurumunun, Hava Kuvvetleri Komutanlığı için teklif iş akışı

### 7.8.3 Satıcı1'in sipariş alma iş akışı

Satıcı1'in sipariş iş akışı Şekil 7.15'te verilmiştir. Buna göre önce S1Siparis kuyruğundan fatura XML dokümanı alınır. S1SiparisV kanalı ile doküman, sipariş verisini Satıcı1'in veritabanına kaydeden ve Hava Kuvvetleri Komutanlığına fatura gönderen porta verilir.. Faturanın Hava Kuvvetleri Komutanlığına gönderilmesi ile Hava Kuvvetleri Komutanlığının mal alım işlemi sona ermektedir.



Şekil 7.15 Satıcı1 kurumunun, Hava Kuvvetleri Komutanlığı için sipariş iş akışı

## 8 UYGULAMA

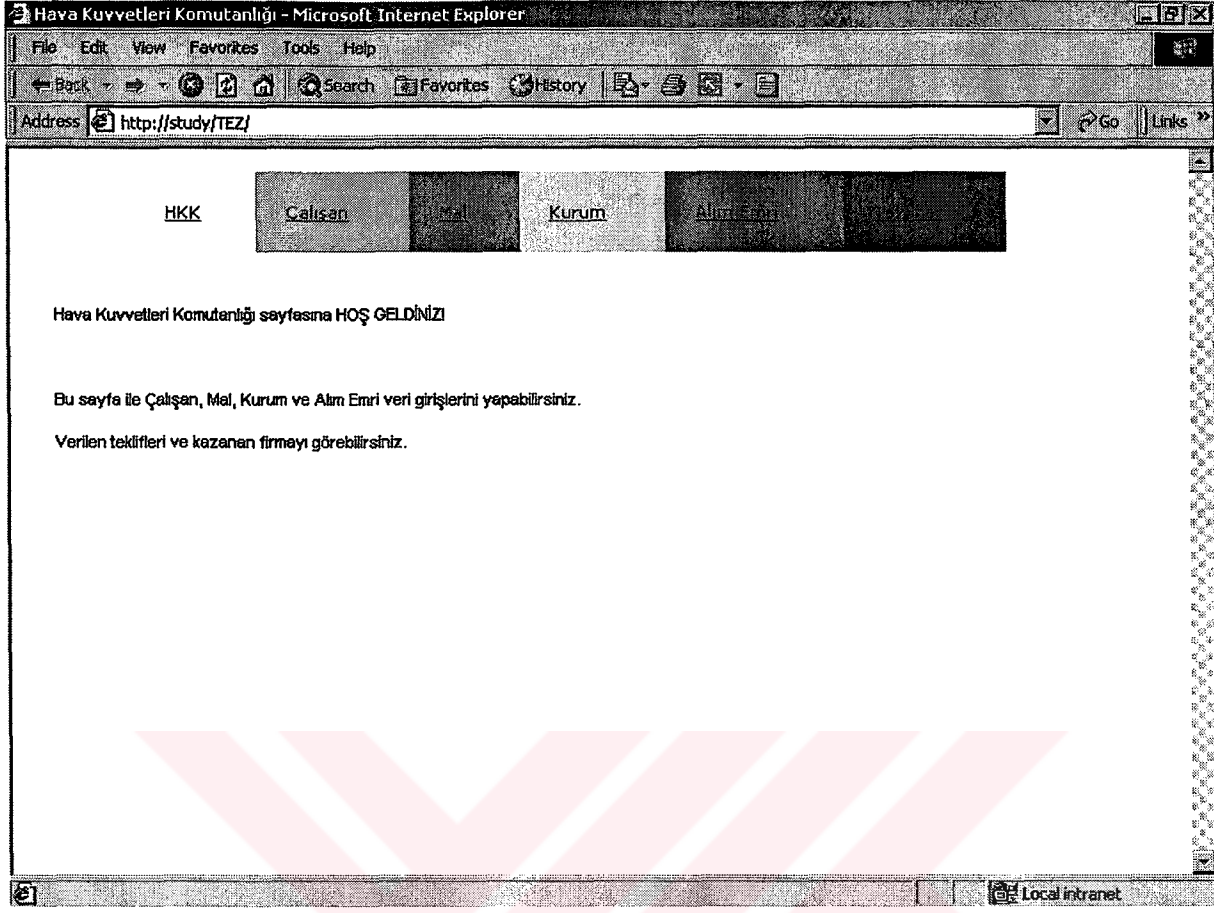
Bu bölümde yedinci bölümde tasarlanan B2B uygulaması için geliştirilmiş olan basit web tabanlı arayüzleri ve iş süreci modelindeki iş akışları açıklanmıştır.

### 8.1 Kullanılan Yazılımlar

Kurumlar arası elektronik ticaret modeli geliştirilirken, Microsoft BizTalk Server 2000 sunucu yazılımı, Microsoft Visual Studio 6.0 uygulama geliştirme yazılımı; Microsoft SQL Server 2000 veritabanı kullanılmıştır. Ayrıca, Microsoft Windows işletim sistemine özel Internet Information Services ve Message Queueing Services yazılım parçalarından da yararlanılmıştır.

### 8.2 Alıcı Kurum Tarafındaki Arayüzü ve İş Süreci

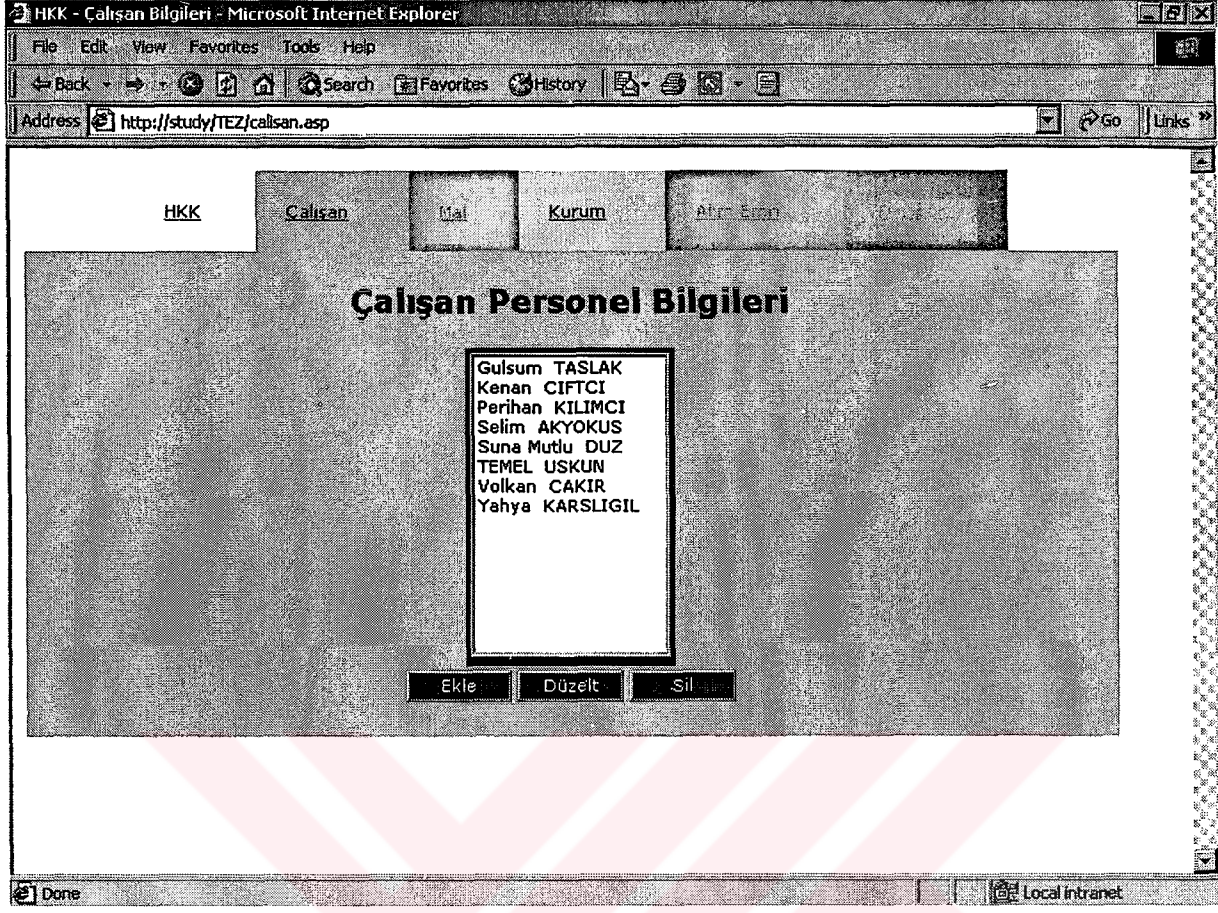
Hava Kuvvetleri Komutanlığı'nın arayüzü Şekil 8.1'de verilmiştir. Bu ekran ile Çalışan, Mal, Kurum ve Alım Emri verileri girilebilmekte ayrıca ve satıcı kurumların verdiği teklifler ve mal alımı için sipariş verilen kurum görülebilmektedir. Satın alınacak mallar, universal kod ile veritabanında saklanmaktadır. Bu yüzden tüm veritabanlarında universal kod ile tüm mallar kayıtlıdır. Ayrıca hangi kurumlar arasında B2B gerçekleşiyorsa bu kurumlar da her satıcının ve alıcının veritabanlarında saklanmaktadır.



Şekil 8.1 Hava Kuvvetleri Komutanlığı arayüzü

### 8.2.1 Çalışan sayfası ile yapılan işlemler

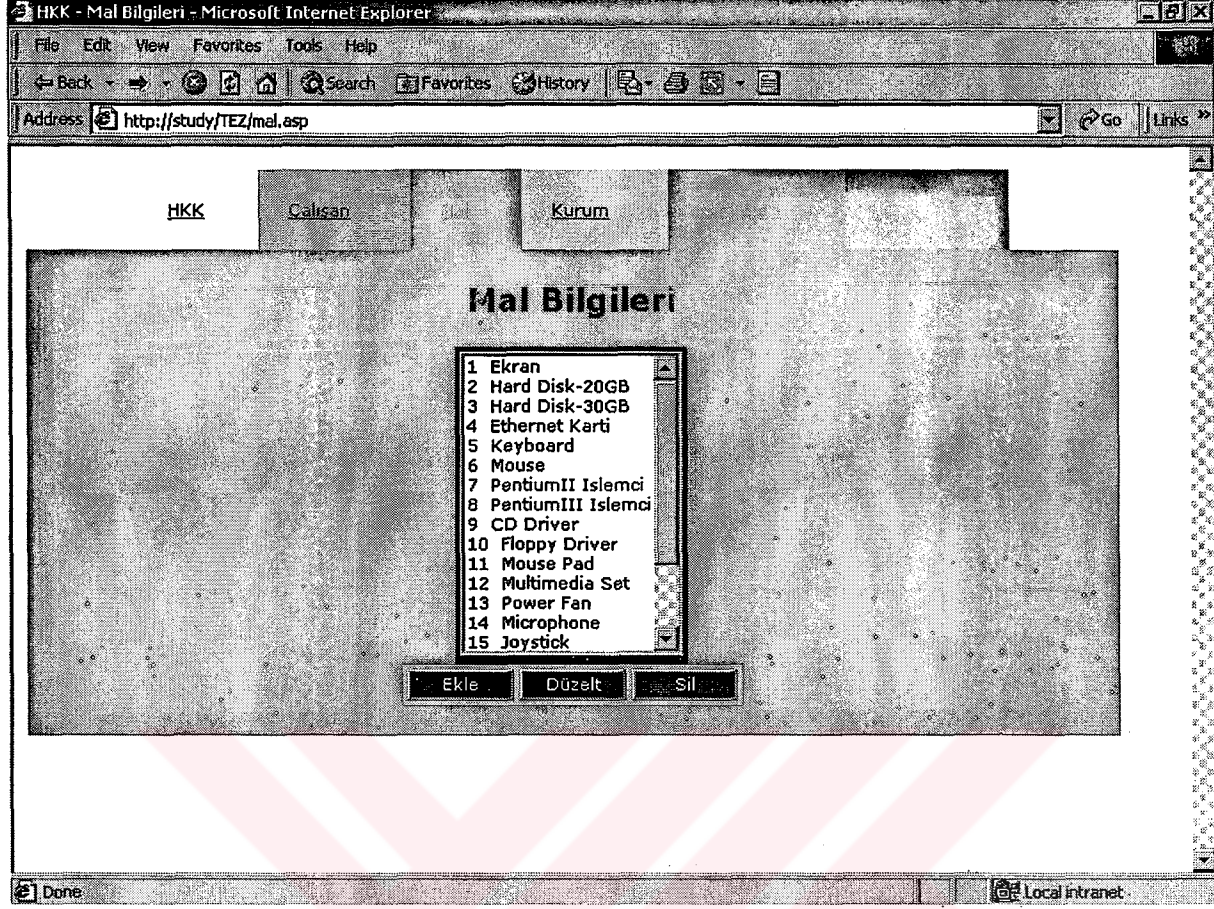
Alım emri isteklerini çalışan personel vermektedir. Verilen emirlerden sorumlu personelin saptanması için, her alım emrini verecek olan personel bilgisi de alım emri isteğine eklenmektedir. Bu nedenle B2B ticaret modelimizde, çalışan verileri de tutulacaktır. Çalışan verilerinin girişinin yapıldığı, değiştirildiği ve silindiği sayfa Şekil 8.2’de verilmiştir. Burada varolan bir kayıt bir daha eklenememekte ve varolan bir kayıt ile veri güncelleştirilememektedir.



Şekil 8.2 Çalışan personel bilgileri sayfası

### 8.2.2 Mal sayfası ile yapılan işlemler

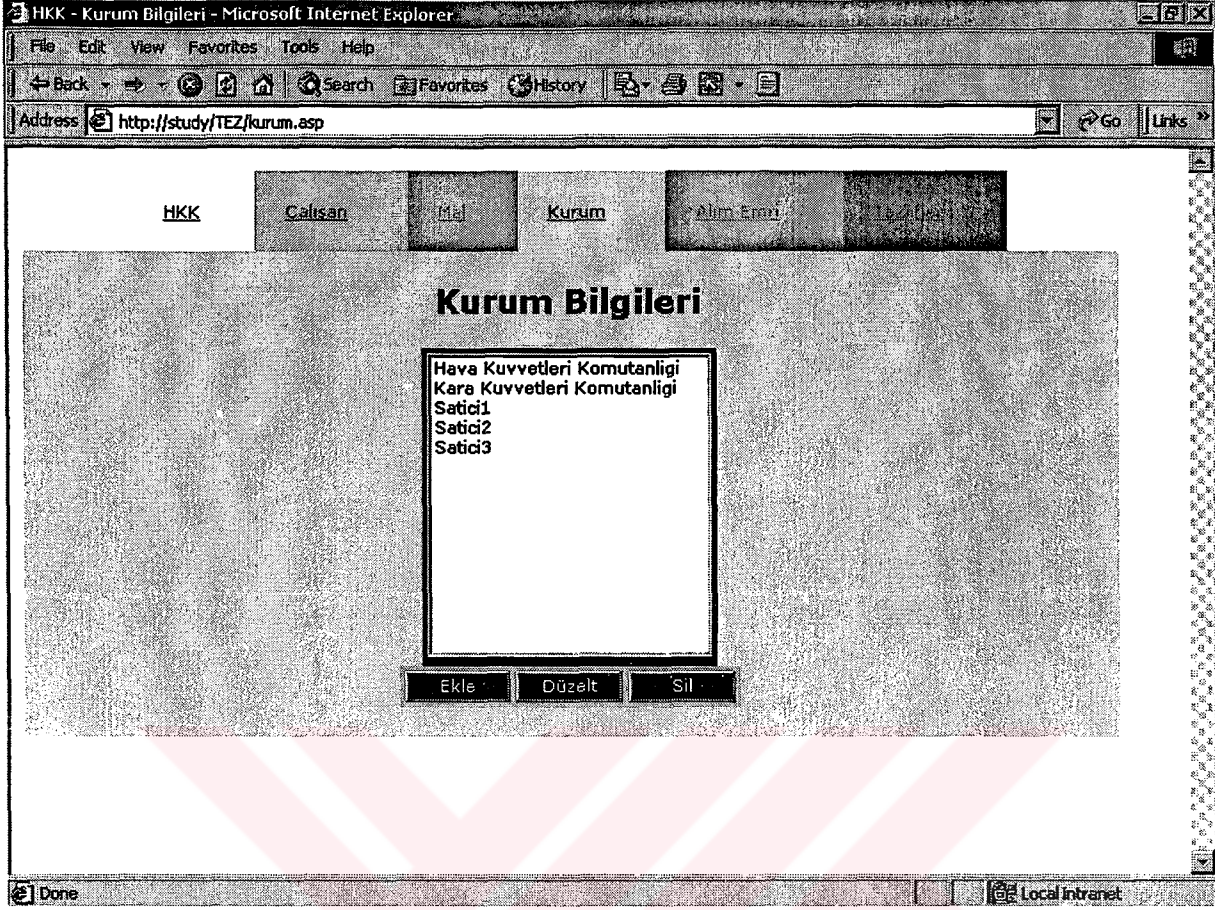
Satılacak mallar dünya üzerinde standart bir kodla kodlanmış olmalıdır ki her satıcı ve alıcı bir kod verdiğinde, bu kodla sadece tek bir mal anlaşılmalıdır. RosettaNet yapısı, bilgisayar parçaları için standart kodlar geliştirmektedir. Bu standart kodlar, yakın zamanda B2B ticaretinde yaygın olarak kullanılacaktır. Bu uygulamayı geliştirirken biz de Hava Kuvvetleri Komutanlığının bilgisayar parçaları satın alması senaryosu üzerinde durduk. Bu verileri HKK, KKK, Satıcı1, Satıcı2, Satıcı3 veritabanlarında aynı şekilde kodladık. Bu verilerin tüm sistemde aynı olması bu uygulama için ön şarttır. Bir mal eklendiğinde bu malın sistemdeki tüm veritabanlarında olması gerekmektedir. Mal verilerinin girişinin yapıldığı, değiştirildiği ve silindiği sayfa Şekil 8.3'te verilmiştir. Çalışan personel sayfasında olduğu gibi, varolan bir kayıt bir daha eklenememekte ve varolan bir kayıt ile veri güncelleştirilememektedir.



Şekil 8.3 Mal bilgileri sayfası

### 8.2.3 Kurum sayfası ile yapılan işlemler

B2B ticareti yapan tüm kurumların verileri her bir kurumun veritabanında saklanmalıdır. Bu sayede mesajın hangi kurumdan geldiği, veritabanından mesajın geldiği kurumun koduna bakılarak anlaşılmaktadır. Bir B2B ticaret modelinde, tüm alıcı kurumların güvendiği kurumlar arasında ticaret gerçekleştirilmektedir. Kurum verilerinin girişinin yapıldığı, değiştirildiği ve silindiği sayfa Şekil 8.4'te verilmiştir.



Şekil 8.4 Kurum bilgileri sayfası

#### 8.2.4 Alım emri işlemi

Alım Emrinin verildiği sayfa, Şekil 8.5'te verilmiştir. Bu sayfa ile alım emri isteği için gerekli veriler ekrandan alınmaktadır. Öncelikle tarih, sistemden alınmaktadır. XML'de tarih verisi yıl, ay, gün şeklinde tutulmaktadır. Daha sonra emri verecek çalışan listeden seçilmektedir. En son olarak da alımı istenen mallar seçilerek adetleri girilmektedir.



HKK - Alım Emri - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Home Search Favorites History Print

Address http://study/TEZ/hkAlimEmri.asp Go Links

HKK Calisan Mal Kurum

## Alım Emri

Tarih: 200-09-23

Alım Emri verecek Çalışan: Yahya KARSLIGIL

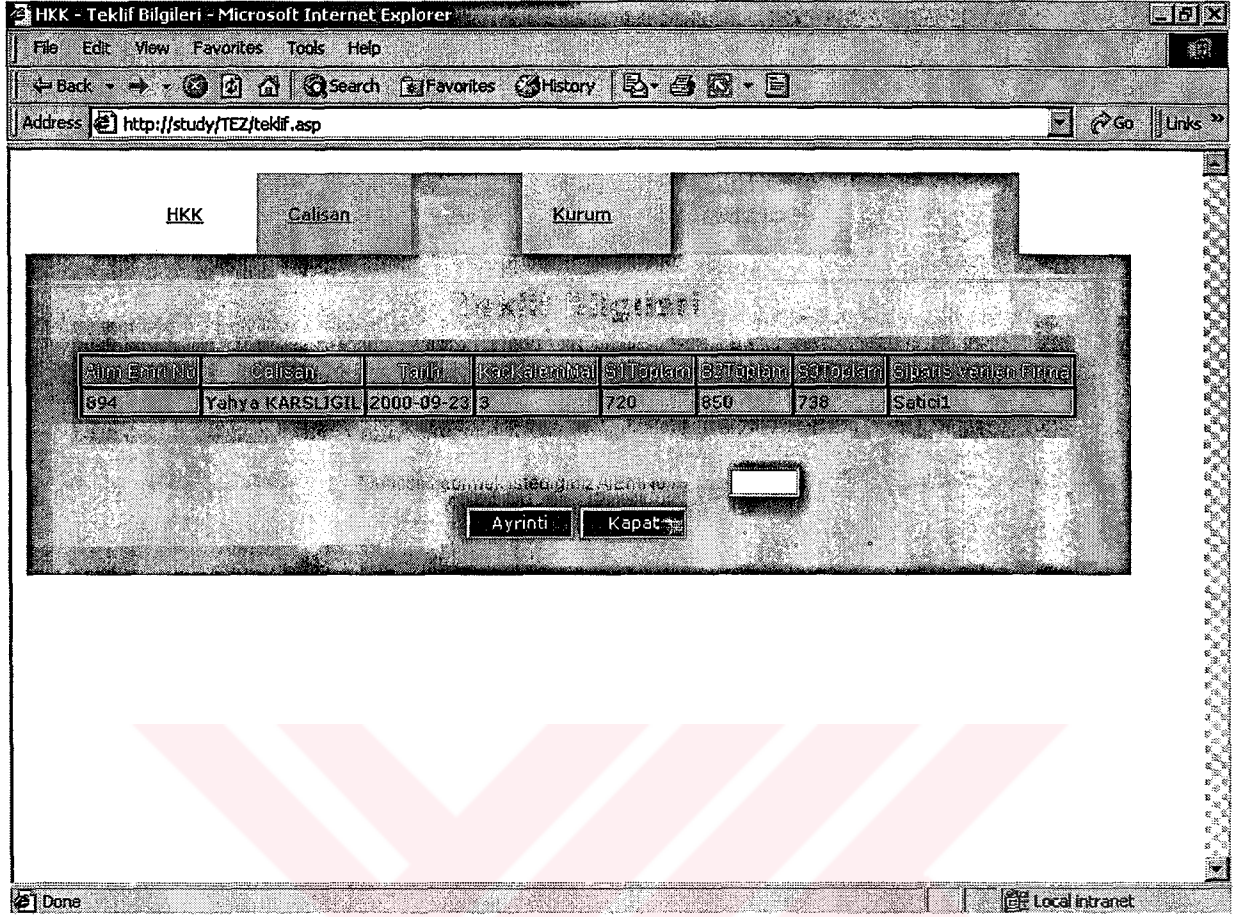
Mal	Adet
Ekran Kartı	2
Floppy Driver	4
Hard Disk-30GB	8

Done Local intranet

Şekil 8.5 Alım emri sayfası

### 8.2.5 Satıcı kurumların verdiği teklifler

Hava Kuvvetleri Komutanlığına satıcı kurumların verdiği teklifler, Şekil 8.6'da verilmiştir. İstenilen alım emrinin ayrıntısı, numarası girilerek görülebilmektedir.

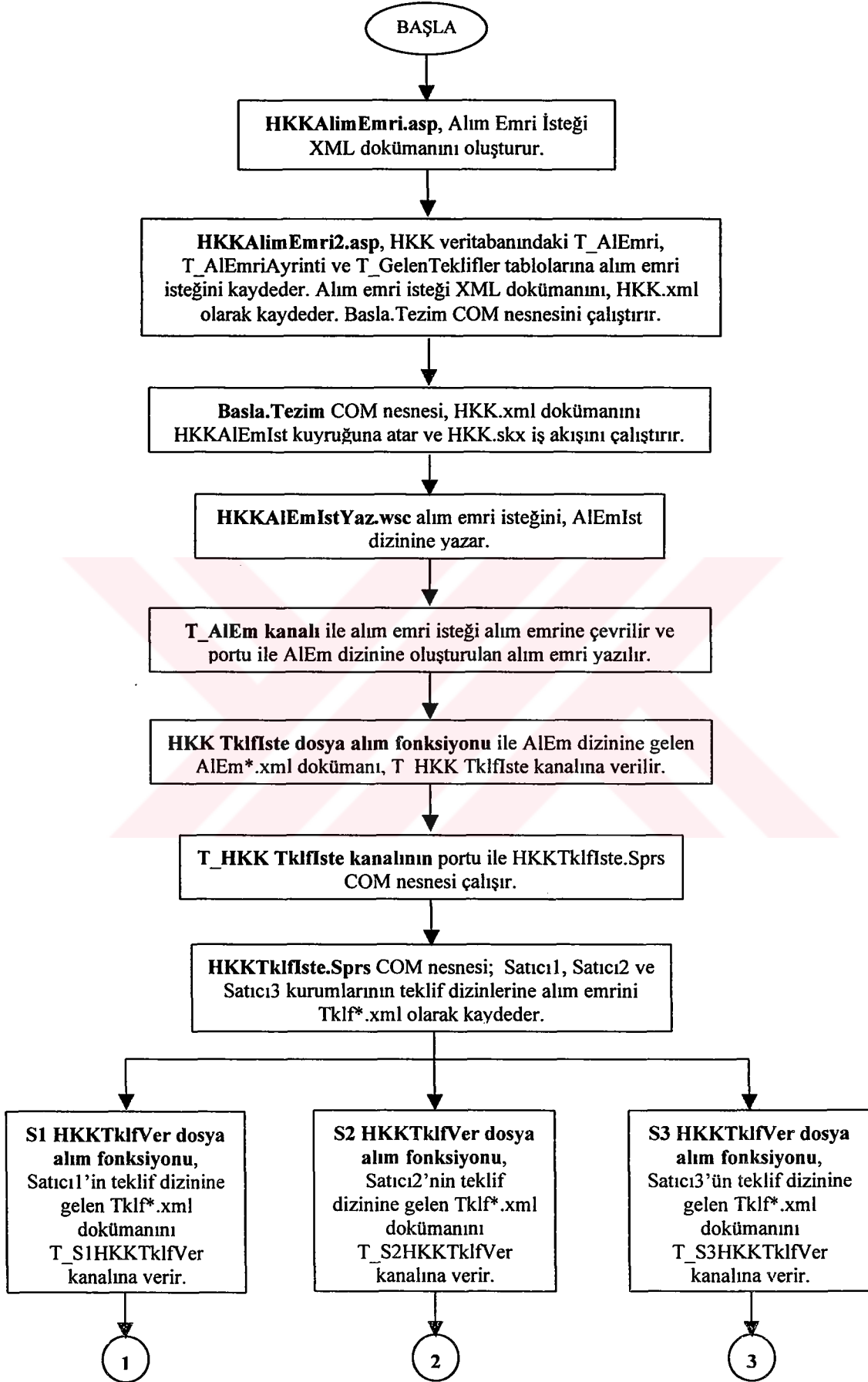


Şekil 8.6 Hava Kuvvetleri Komutanlığına verilen teklifler

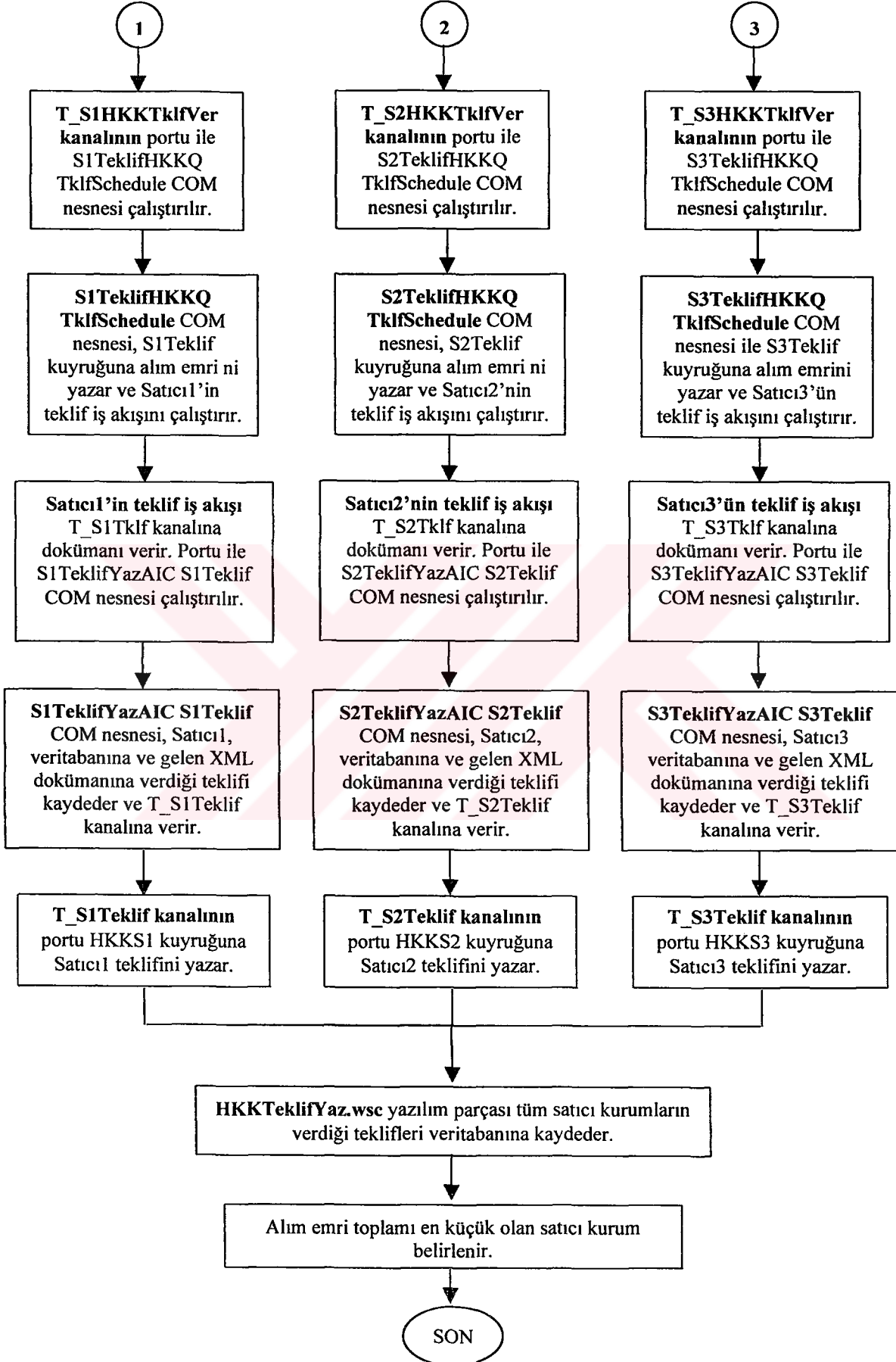
### 8.3 Mal Alımında Alıcı-Satıcı Tarafında Çalışan Uygulamalar

Alım emri isteği yaratıldıktan sonra Ek 10'da verilen dosya alım fonksiyonları ve BTSInterchangeLib.Interchange nesnesinin Submit metodu ile önce kanallar daha sonra da portlar etkinleştirilerek XML mesajının yaratıldığı yerden gitmesi gereken yere gönderilmesi gerçekleştirilmektedir. Kurumlar arası ticaret modelimizi geliştirirken port isimlerinin "T\_Port" harfleri ile başlaması ve kanal isimlerinin port isimleri ile aynı olması fakat "T\_" harfleri ile başlaması bir standart olarak kullanılmıştır.

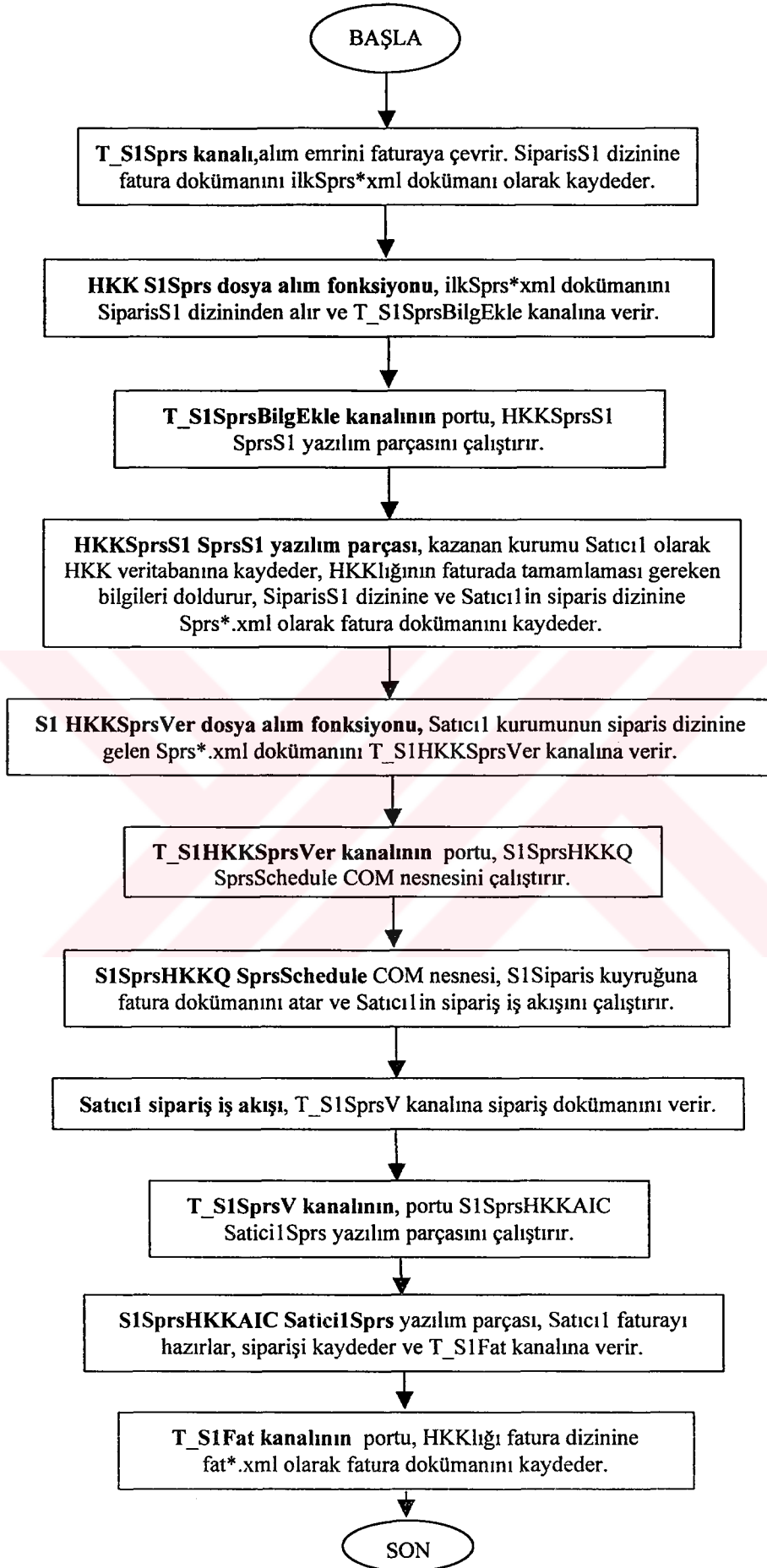
Alım emri isteği yaratılmasından kazanan kurumun belirlenmesine kadar olan aşamada, çalışan programlar ve açıklamalar Şekil 8.7a, Şekil 8.7b'de verilen akış şeması ile gösterilebilir. Kazanan kuruma sipariş verildikten HKKlığına fatura gönderimine kadar olan aşamada, çalışan uygulamalar ve açıklamalar Şekil 8.8'de verilen akış şeması ile gösterilebilir. Bu akış şeması için kazanan kurumun Satıcı1 olduğu varsayılmaktadır.



Şekil 8.7a HKKlığının teklif alımı sırasında çalışan uygulamalar



Şekil 8.7b HKKlığının teklif alımı sırasında çalışan uygulamalar



Şekil 8.8 HKKlığının sipariş verip, fatura alımına kadar çalışan uygulamalar

#### 8.4 Mesajlaşmada Kullanılan XML Dokümanları

Alım Emri ekranından “Alım Emrini Ver” düğmesine basılarak Alım Emri işlemi başlatılmaktadır. Öncelikle, ekrandan alınan bilgilerle alım emri isteği XML dokümanı oluşturulmaktadır. Oluşturulan XML dokümanı, Şekil 8.9’da verilmiştir.

```
<AlEmIst>
  <Baslik Tarih="2000-09-23" AlEmNo="894" KurumNo="1" />
  <CalisanBilgisi Soyad="KARSLIGIL" Ad="Yahya" CalisanNo="41" />
  <Mal Tanim="Ekran Karti" UniversalKod="1" OlcuBirimi="EA"
  Adet="2" SatirNo="1" />
  <Mal Tanim="Floppy Driver" UniversalKod="10" OlcuBirimi="EA"
  Adet="4" SatirNo="2" />
  <Mal Tanim="Hard Disk-30GB" UniversalKod="3" OlcuBirimi="EA"
  Adet="8" SatirNo="3" />
  <Toplam KacKalemMal="3" />
</AlEmIst>
```

Şekil 8.9 Alım emri isteği XML dokümanı

Alım emri isteğinin hangi verilerden oluşabileceğine ait bilgi, WebDAV’daki T\_AlEmIst.xml dokümanından alınmaktadır. Alım emri isteği onaylandıktan sonra alım emri XML dokümanına çevrilmektedir. Oluşturulan alım emri ile satıcı kurumlardan teklif istenmektedir. Satıcı1’in gönderdiği teklif bilgileri, Hava Kuvvetleri Komutanlığına Şekil 8.10’daki gibi bir XML dokümanı ile gönderilir.

```
<AlEm>
  <Baslik Tarih="2000-09-23" AlEmNo="894" KurumNo="3" />
  <CalisanBilgisi Soyad="KARSLIGIL" Ad="Yahya" CalisanNo="41"
  />
  <Mal Tanim="Ekran Karti" UniversalKod="1" OlcuBirimi="EA"
  Adet="2" Fiyat="80" HesaplanmisFiyat="160" SatirNo="1" />
  <Mal Tanim="Floppy Driver" UniversalKod="10" OlcuBirimi="EA"
  Adet="4" Fiyat="80" HesaplanmisFiyat="320" SatirNo="2" />
  <Mal Tanim="Hard Disk-30GB" UniversalKod="3" OlcuBirimi="EA"
  Adet="8" Fiyat="30" HesaplanmisFiyat="240" SatirNo="3" />
  <Toplam KacKalemMal="3" AlEmToplami="720" />
</AlEm>
```

Şekil 8.10 Alım emri XML dokümanı

Satıcı kurumlar, alım emri dokümanındaki mallara verdikleri fiyatları, hesaplanmış fiyatları ve alım emri toplamı alanlarını doldurmaktadır. Tüm satıcı kurumlardan teklifler geldikten sonra verilen tekliflerden en düşük AlEmToplami olan kuruma sipariş verilmektedir. Yukarıdaki örnek için, en düşük toplam fiyatı Satıcı1 vermiştir. Bu yüzden Hava Kuvvetleri Komutanlığı mal alımını Satıcı1 kurumundan yapacaktır. Satıcı1 kurumu, kazanan kurum

olarak belirlendikten sonra Satıcı1, alım emrini faturaya çevirir. Faturanın hangi verilerden oluşabileceğine ait bilgi, WebDAV'daki T\_Fatura.xml dokümanından alınmaktadır. Kazanan kurum belirlendikten sonra fatura dokümanı Satıcı1 kurumuna gitmeden aşağıdaki şekliyle S1Siparis dizinine kaydedilmektedir. Satıcı1 kurumunun siparişi işlediğinin anlaşılması için alıcı bilgilerinin tamamı doldurulmamıştır. Satıcı bilgilerini de Satıcı1 kurumunun doldurması gerektiğinden, bu alan tamamen boştur. Sipariş aşamasında fatura XML dokümanı Şekil 8.11'de verilmiştir.

```
<Fatura>
  <Baslik Tarih="" Numara="" ALEmNo="894" ALEmTarih="2000-09-23"
  GondermeTarihi="" />
  <Satıcı>
    <Adres Ad="" Adres1="" Adres2="" Il="" PostaKodu="" />
    <KontaktBilgisi Ad="" CalisanNo="" />
  </Satıcı>
  <Alici>
    <Adres Ad="" Adres1="" Adres2="" Il="" PostaKodu="" />
    <KontaktBilgisi Ad="Yahya KARSLIGIL" CalisanNo="41" />
  </Alici>
  <FaturaSatisSartlari IndirimOrani="" />
  <Mal Tanim="Ekran Karti" UniversalKod="1" OlcuBirimi="EA" Adet="2" Fiyat="80"
  HesaplanmisFiyat="160" SatirNo="1" />
  <Mal Tanim="Floppy Driver" UniversalKod="10" OlcuBirimi="EA" Adet="4"
  Fiyat="80" HesaplanmisFiyat="320" SatirNo="2" />
  <Mal Tanim="Hard Disk-30GB" UniversalKod="3" OlcuBirimi="EA" Adet="8"
  Fiyat="30" HesaplanmisFiyat="240" SatirNo="3" />
  <Toplam KacKalemMal="3" ALEmToplami="720" IndirimMiktari="" VergiMiktari=""
  FaturaToplami="" />
  <FaturaAdresi>
    <Adres Ad="Hava Kuvvetleri K.ligi" Adres1="Merkez" Adres2="" Il="ANKARA"
    PostaKodu="06000" />
  </FaturaAdresi>
  <GondermeAdresi>
    <Adres Ad="Hava Harp Okulu K.ligi" Adres1="Yesilyurt" Adres2=""
    Il="ISTANBUL" PostaKodu="34800" />
  </GondermeAdresi>
</Fatura>
```

Şekil 8.11 Sipariş aşamasında fatura XML dokümanı

Yukarıdaki şekilde sipariş verilerini alan Satıcı1 kurumu sipariş kaydı yapmakta ve hazırladığı faturayı Hava Kuvvetleri Komutanlığı'na göndermektedir. Satıcı1 kurumunun gönderdiği fatura Şekil 8.12'de verilmiştir.

```

<Fatura>
  <Baslik Tarih="2000-09-23" Numara="548" AEmNo="894" AEmTarih="2000-09-23"
  GondermeTarihi="2000-09-23" />
  <Satici>
    <Adres Ad="Satici1" Adres1="Merkez" Adres2="" Il="IZMIR" PostaKodu="35000" />
    <KontaktBilgisi Ad="Satici1 Calisani" CalisanNo="1245" />
  </Satici>
  <Alici>
    <Adres Ad="Hava Kuvvetleri K.ligi" Adres1="Merkez" Adres2="" Il="ANKARA"
    PostaKodu="06000" />
    <KontaktBilgisi Ad="Yahya KARSLIGIL" CalisanNo="41" />
  </Alici>
  <FaturaSatisSartlari IndirimOrani="10" />
  <Mal Tanim="Ekran Karti" UniversalKod="1" OlcuBirimi="EA" Adet="2" Fiyat="80"
  HesaplanmisFiyat="160" SatirNo="1" />
  <Mal Tanim="Floppy Driver" UniversalKod="10" OlcuBirimi="EA" Adet="4" Fiyat="80"
  HesaplanmisFiyat="320" SatirNo="2" />
  <Mal Tanim="Hard Disk-30GB" UniversalKod="3" OlcuBirimi="EA" Adet="8" Fiyat="30"
  HesaplanmisFiyat="240" SatirNo="3" />
  <Toplam KacKalemMal="3" AEmToplami="720" IndirimMiktari="36" VergiMiktari="72"
  FaturaToplami="756" />
  <FaturaAdresi>
    <Adres Ad="Hava Kuvvetleri K.ligi" Adres1="Merkez" Adres2="" Il="ANKARA"
    PostaKodu="06000" />
  </FaturaAdresi>
  <GondermeAdresi>
    <Adres Ad="Hava Harp Okulu K.ligi" Adres1="Yesilyurt" Adres2="" Il="ISTANBUL"
    PostaKodu="34800" />
  </GondermeAdresi>
</Fatura>

```

Şekil 8.12 Fatura aşamasında fatura XML dokümanı

Başlık verisindeki numara, Satıcı kurumunun verdiği teklif numarasıdır. Bu fatura tutarı hesaplanırken, %10 vergi eklenmiş ve %5 indirim yapılmıştır.

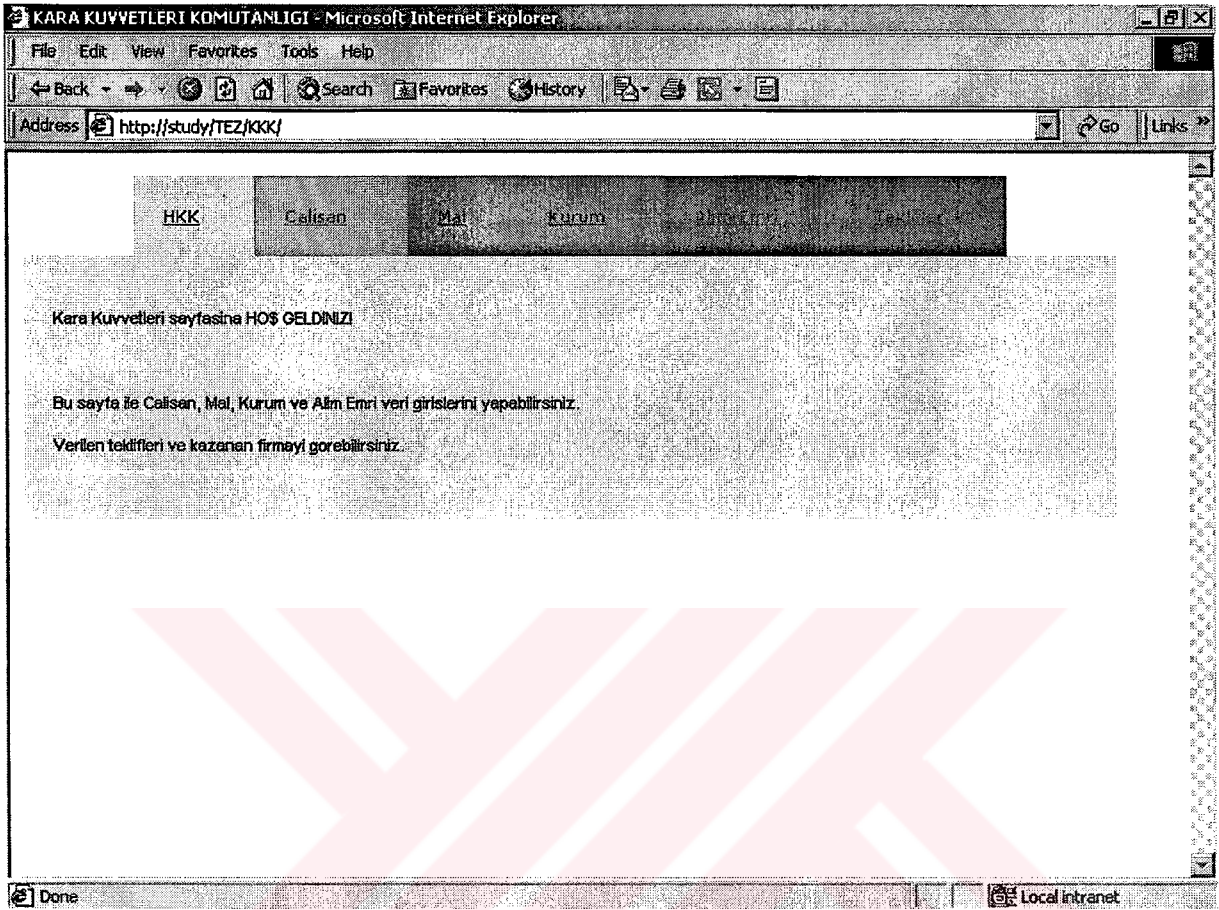
HKKlğının yaptığı tüm işlemler, logHKK.txt dosyasına kaydedilmektedir.

## 8.5 İkinci Alıcı Kurum Tarafındaki Arayüzü ve İş Süreci

Kara Kuvvetleri Komutanlığı'nın arayüzü Şekil 8.13'te verilmiştir. Bu ekran ile Hava Kuvvetleri sayfasında olduğu gibi, Çalışan, Mal, Kurum ve Alım Emri verileri girilebilir ve Satıcı kurumların verdiği teklifler ve mal alımı için sipariş verilen kurum görülebilmektedir. Alım emri senaryosu, Hava Kuvvetleri Komutanlığı sayfasında belirtildiği gibi gerçekleştirilmektedir.



KKKlğının yaptıđı tđm iřlemler, logKKK.txt dosyasına kaydedilmektedir.



Őekil 8.13 Kara Kuvvetleri Komutanlıđı arayđzđ

## 8.6 Birinci Satıcı Kurum Tarafındaki Arayđzđ ve İř Sđreci

Satıcı1 arayđzđ ile Satıcı1'in sattıđı mallar, verdiđi teklifler ve verilen sipariřler gđrđlebilmektedir. Satıcı1 kurumundan istenen teklifleryani kazanılmayan teklifler, Őekil 8.14'te verilmiřtir. Satıcı1 kurumuna verilen sipariřleryani kazanılan teklifler ise Őekil 8.15'te verilmiřtir.

SATICI1 - Teklif Bilgileri - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Home Search Favorites History

Address http://study/TEZ/satici1/teklif.asp Go Links

Satici1 Mal Teklif Siparis

### Teklif Bilgileri

Teklif No	Alım Emri No	Tarih	Çalışan	Kaç Kalem Mal	Teklif Toplamı	Teklif İsteyen Kurum
754	192	2000-10-29	Murat ERMIS	2	870	Kara Kuvvetleri Komutanlığı
755	1026	2000-10-29	Yahya KARSLIGIL	2	500	Hava Kuvvetleri Komutanlığı
756	1027	2000-10-29	Suna Mutlu DUZ	3	5845	Hava Kuvvetleri Komutanlığı
758	193	2000-10-29	Elif KARSLIGIL	2	1190	Kara Kuvvetleri Komutanlığı
763	1030	2000-10-29	Gulsum TASLAK	1	680	Hava Kuvvetleri Komutanlığı
764	1031	2000-10-29	TEMEL USKUN	2	2480	Hava Kuvvetleri Komutanlığı
766	1033	2000-10-29	TEMEL USKUN	2	830	Hava Kuvvetleri Komutanlığı
767	1034	2000-10-29	Yahya KARSLIGIL	5	1451	Hava Kuvvetleri Komutanlığı
768	1035	2000-10-29	Volkan CAKIR	3	524	Hava Kuvvetleri Komutanlığı
769	1036	2000-10-29	TEMEL USKUN	1	180	Hava Kuvvetleri Komutanlığı
772	198	2000-10-29	Elif KARSLIGIL	2	375	Kara Kuvvetleri Komutanlığı
773	199	2000-10-29	Murat ERMIS	1	6	Kara Kuvvetleri Komutanlığı
774	200	2000-10-29	Murat ILGAZ	1	585	Kara Kuvvetleri Komutanlığı
775	201	2000-10-29	Murat ERMIS	1	120	Kara Kuvvetleri Komutanlığı
776	203	2000-10-29	Murat ERMIS	1	2	Kara Kuvvetleri Komutanlığı
777	204	2000-10-29	Mustafa ER	1	510	Kara Kuvvetleri Komutanlığı
778	205	2000-10-29	Oguzhan ERDINC	1	480	Kara Kuvvetleri Komutanlığı

Done Local intranet

Şekil 8.14 Satıcı1 teklif sayfası

Satıcı1 kurumunun yaptığı tüm işlemler, logSatici1.txt dosyasına kaydedilmektedir.

SATICI1 - Sipariş Bilgileri - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Search Favorites History

Address http://study/TEZ/satici1/siparis.asp

Saticı1 Mal Teklif Sipariş

### Sipariş Bilgileri

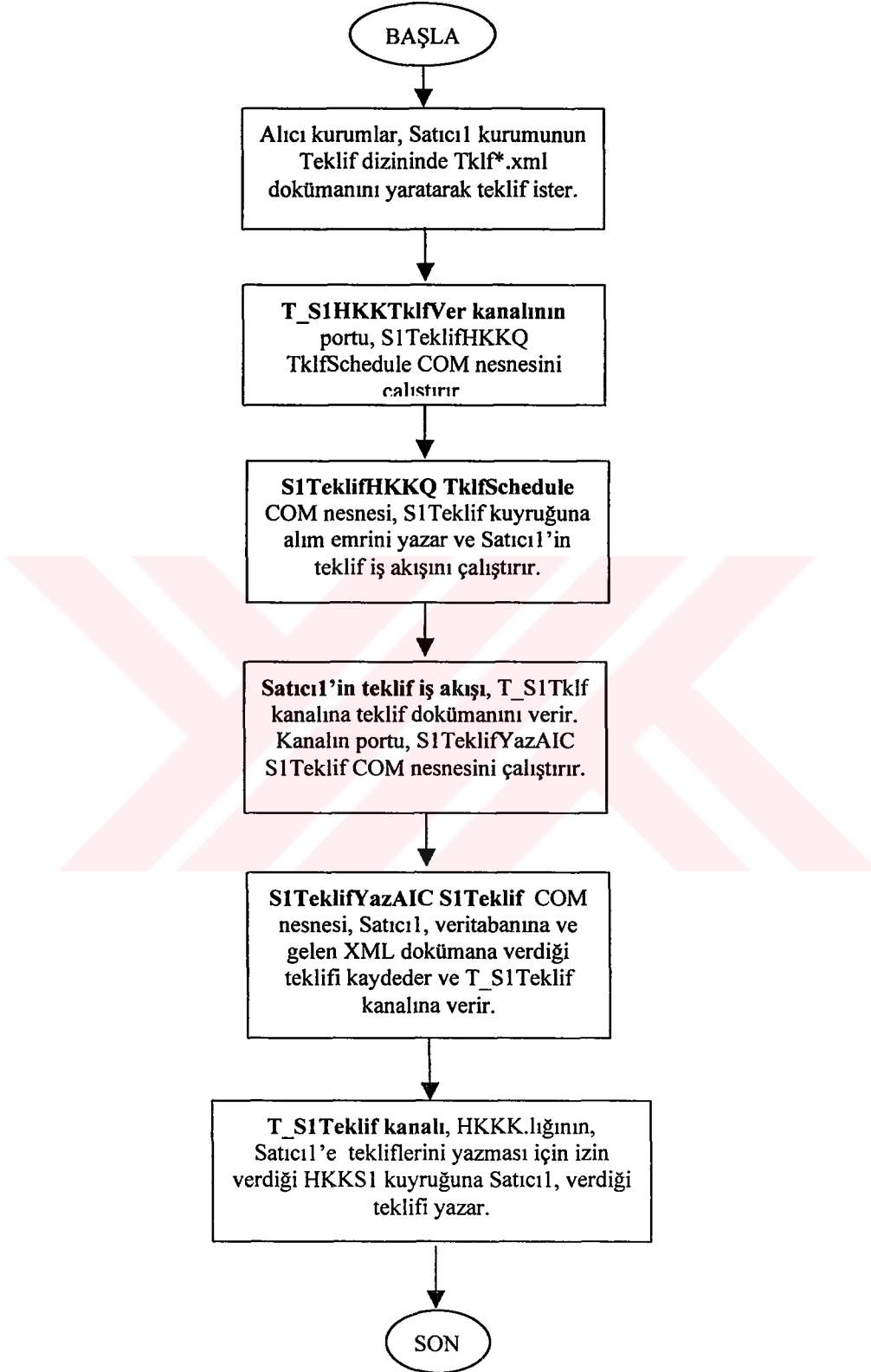
Teklif No	Alım Emri No	Tarih	Çalışan	Kaç Kalem Mal	Teklif Toplamı	Sipariş Veren Kurum
753	1025	2000-10-29	Selim AKYOKUS	2	340	Hava Kuvvetleri Komutanlığı
757	1029	2000-10-29	Kenan CIFTCI	4	225	Hava Kuvvetleri Komutanlığı
759	194	2000-10-29	Mustafa ER	3	1740	Kara Kuvvetleri Komutanlığı
760	195	2000-10-29	Oguzhan ERDINC	3	3245	Kara Kuvvetleri Komutanlığı
761	196	2000-10-29	Murat ILGAZ	2	240	Kara Kuvvetleri Komutanlığı
762	197	2000-10-29	Murat ERMIS	1	480	Kara Kuvvetleri Komutanlığı
765	1032	2000-10-29	Perihan KILIMCI	3	406	Hava Kuvvetleri Komutanlığı
770	1037	2000-10-29	Perihan KILIMCI	2	646	Hava Kuvvetleri Komutanlığı
771	1038	2000-10-29	Selim AKYOKUS	1	150	Hava Kuvvetleri Komutanlığı
784	212	2000-10-29	Murat ILGAZ	1	1350	Kara Kuvvetleri Komutanlığı
787	1041	2000-10-29	Yahya KARSLIGIL	6	3670	Hava Kuvvetleri Komutanlığı
788	1042	2000-10-30	TEMEL USKUN	1	300	Hava Kuvvetleri Komutanlığı
790	1044	2000-10-30	Yahya KARSLIGIL	1	280	Hava Kuvvetleri Komutanlığı
793	215	2000-10-30	Oguzhan ERDINC	1	960	Kara Kuvvetleri Komutanlığı
796	1047	2000-10-30	Kenan CIFTCI	3	1150	Hava Kuvvetleri Komutanlığı
798	1049	2000-10-30	Selim AKYOKUS	1	150	Hava Kuvvetleri Komutanlığı
807	1053	2000-10-30	Perihan KILIMCI	1	480	Hava Kuvvetleri Komutanlığı

Done Local intranet

Şekil 8.15 Saticı1 sipariş sayfası

### 8.6.1 Saticı1 kurumundan teklif istendiğinde çalışan uygulamalar

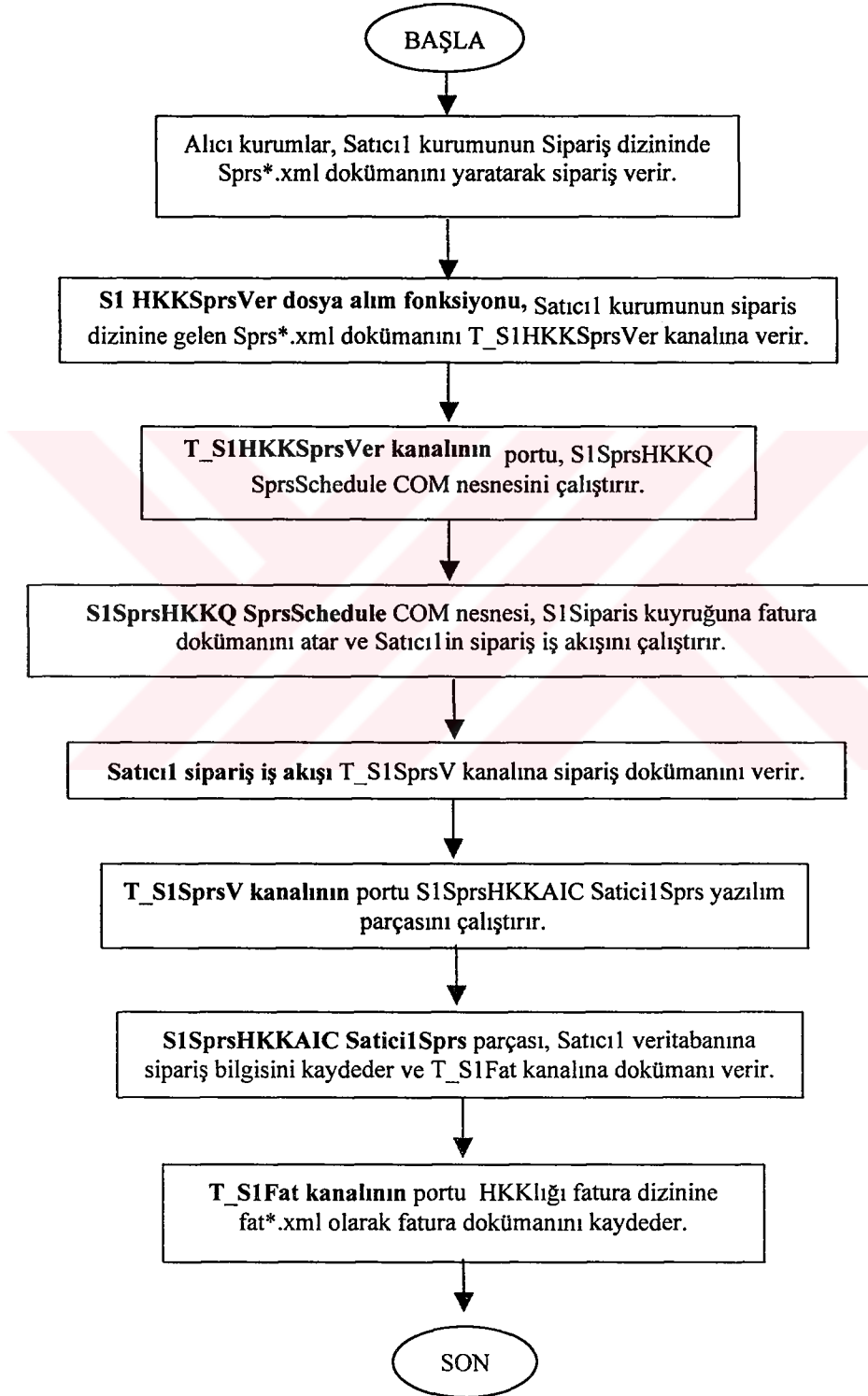
Saticı1 kurumundan teklif istendiğinde, çalışan uygulamalar ve açıklamalar Şekil 8.16'da verilen akış şeması ile gösterilebilir.



Şekil 8.16 Satıcı1 kurumundan teklif istendiğinde çalışan uygulamalar

### 8.6.2 Satıcı1 kurumuna sipariş verildiğinde çalışan uygulamalar

Satıcı1 kurumuna sipariş verildiğinde çalışan uygulamalar ve açıklamalar aşağıda verilen akış şeması ile gösterilebilir.

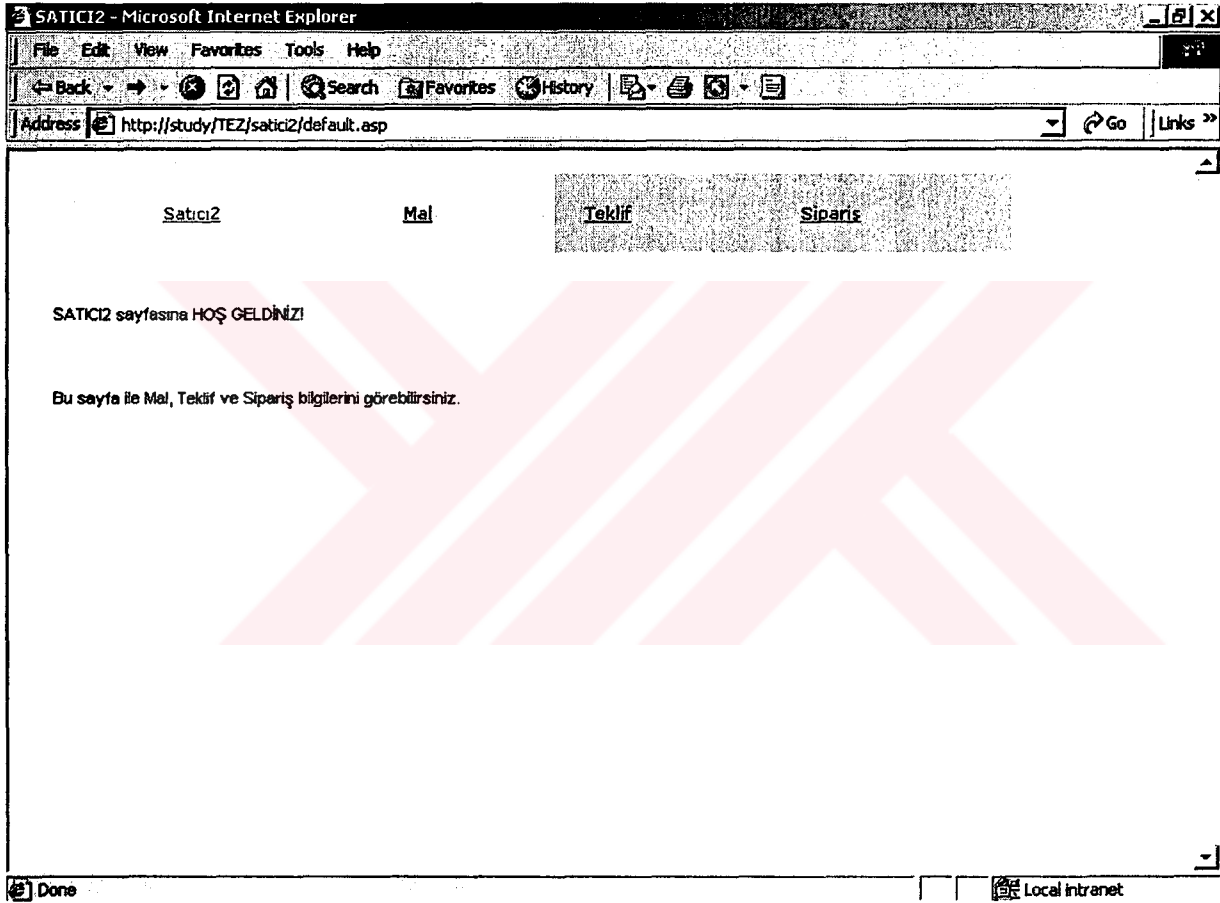


Şekil 8.17 Satıcı1 kurumuna sipariş verildiğinde çalışan uygulamalar

## 8.7 İkinci Satıcı Kurum Tarafındaki Arayüzü ve İş Süreci

Satıcı2 arayüzü ile Satıcı2'nin sattığı mallar, verdiği teklifler ve verilen siparişler görülebilmektedir. Satıcı2 kurumunun sayfası Şekil 8.18'de verilmiştir.

Satıcı2 kurumunun yaptığı tüm işlemler, logSatıcı2.txt dosyasına kaydedilmektedir.

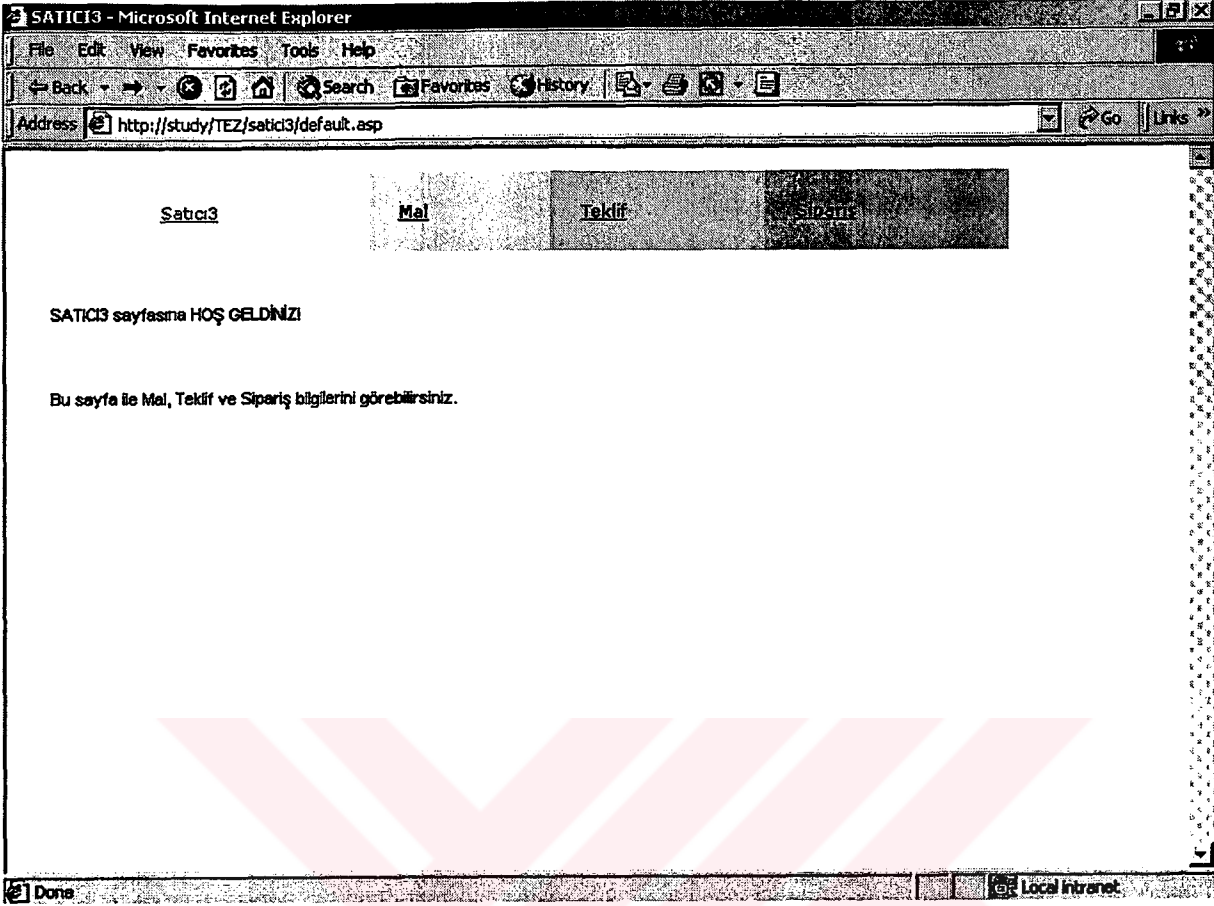


Şekil 8.18 Satıcı2 web sayfası

## 8.8 Üçüncü Satıcı Kurum Tarafındaki Arayüzü ve İş Süreci

Satıcı3 arayüzü ile Satıcı3'ün sattığı mallar, verdiği teklifler ve verilen siparişler görülebilmektedir. Satıcı3 kurumunun sayfası Şekil 8.19'da verilmiştir.

Satıcı3 kurumunun yaptığı tüm işlemler, logSatıcı3.txt dosyasına kaydedilmektedir.



Şekil 8.19 Satıcı3 web sayfası

## 9 SONUÇLAR VE ÖNERİLER

Ortak iş ve ticaret yapan kurumlar, farklı endüstriler farklı bilgi sistemlerini kullanmaktadır. Bu durumda B2B elektronik ticaretin gerçekleştirilmesi için kurumlar, aralarında geçen mesajlaşmayı ve veri işleme tekniklerini standartlaştırmak zorundadır. Kurumlar arası uyumluluğu sağlayabilmek için uluslararası kurumlar bir çok standart geliştirmiştir. Bu standartlar arasında XML-tabanlı B2B yapıları; eCom, RosettaNet ve BizTalk bu tez ile incelenmiştir.

Bu tez çalışması ile elektronik ticaret birçok açıdan araştırılmıştır. İnternet ile elektronik ticaret konusunda yapılan çok fazla sayıda makale incelenmiştir. Haziran 2000'de çıkan ve yeni bir kavram olması nedeniyle; SOAP, WSDL, UDDI teknolojileri üzerine kurulmuş olan web servisleri açıklanmıştır.

Basit bir kurumlar arası elektronik ticaret modeli oluşturulmuş, kurumlar arası uygulama geliştirmede ortaya çıkabilecek güçlükler görülmüş, iş süreçleri modellenmiş, mesajlaşma yapıları tanımlanmış ve alıcı-satıcı kurumlar arasında gerçekleşen elektronik ticaret sistemi yeni çıkan Microsoft BizTalk 2000 Server kullanılarak geliştirilmiştir. Geliştirilen uygulamada iki alıcı kurum olarak, Hava Kuvvetleri Komutanlığı ve Kara Kuvvetleri Komutanlığı; satıcı kurumlar olarak Satıcı1, Satıcı2 ve Satıcı3 kurumları seçilmiştir. Bu sistemde, alıcı kurumun yapabileceği işlemler; alım emri isteğini oluşturmak, alım emri isteğini alım emrine çevirmek, oluşturulan alım emrini satıcı kurumlara göndermek; satıcı kurumlardan gelen teklifleri kaydetmek, bu teklifler arasında en ekonomik olanı seçen alım kararına göre uygun satıcı kuruma sipariş vermek, siparişi kazanan kuruma göndermektir. Satıcı kurumun yapabileceği işlemler ise; teklif oluşturmak, oluşturulan teklifleri alıcı kurumlara göndermek, sipariş almak, fatura göndermektir.

Bu tez ile B2B uygulamalarını geliştirmenin, gerek kullanılan çok farklı teknolojiler, gerekse B2B uygulamalarındaki iş süreçlerinin karmaşıklığı nedeniyle zor bir iş olduğu görülmüştür. Bir çok kurum, kendi B2B uygulama geliştirme araçlarıyla bir iki ayda B2B çözümleri geliştirileceğini söylemektedir ki bu konunun doğruluğu tartışma konusudur. Bir B2B uygulamasındaki, en basit bir sipariş işlemi bile 10-20 adet ek iş sürecinin otomasyonunu gerektirmektedir.



B2B uygulaması geliřtirmek için ilk olarak, seçilen B2B geliřtirme araçlarının ve bu B2B ortamındaki teknolojilerin öğrenilmesi gerekmektedir. İkinci olarak da B2B uygulamalarındaki karmařık iş süreçlerinin etkili bir şekilde otomasyonunu, kontrolünü ve yönetimini sağlayacak iş modelleri ve yazılımlar geliřtirilmelidir. Bu nedenlerle, B2B uygulamalarının başarısının karmařık iş süreçlerini işleyecek çözümler ve bu alanda kullanılan temel teknolojilerin iyi anlaşılmasına baėlı olduėu görülmüřtür.

Yapılan bu tez çalışması doėrultusunda ileride, kurumlar arasındaki mesajlaşmayı sağlayan XML şemaları Türkiye için geliřtirilebilir. Ayrıca, universal mal katalogu kod sunucusu yapılabilir. Bu şemalar ve kataloglar günümüzde olmadığı için yaptığımız tezde, bu teknolojiler tasarlanmış ve gerçekleştirilmiştir.

Satıcı ve alıcı kurumlar için Microsoft BizTalk Server 2000'de tanımladığımız servisler UDDI'ye kaydedilebilir. Böylece, kurumların tanımladığı servisler internet ile tüm kullanıcılara sunulabilir.

Sadece, Microsoft BizTalk Server 2000 ürününün sunduėu eşleyici, internet üzerinde olası ticari ortaklar bulunduktan sonra, otomatik olarak gerçekleştirilebilir.

Çoklu-alıcı ve çoklu-satıcının olduėu bir pazaryeri yaratılabilir ve çeřitli antlaşma yapılarına göre en kısa sürede ve en iyi pazarlığı yapan antlaşma tipi belirlenebilir.

**KAYNAKLAR**

Akyokuş, S., (2001), "XML Tabanlı B2B Çözümleri", Bilişim 2001.

Akyokuş, S., (2000), "XML ve XML Uygulamaları", Bilişim 2000.

Arpınar, S., Dogaç, A., Tatbul, N., (2000), "An Open Electronic Marketplace through Agent-Based Workflows:MOPPET", Intl. Journal on Digital Libraries, Vol. 3, No. 1, July 2000, 36-59.

Baartse, M., Blair, R., v.d., (2000), Professional ASP XML, Wrox Press, Kanada.

Bozkurt, V., (2000), Elektronik Ticaret, Alfa Basım Yayım Dağıtım Ltd. Şti., İstanbul.

Brett, M., (2000), Java and XML, O'Reilly & Associates, Sebastopol.

Chen, Q., Hsu, M., Dayal U., Griss, M., (1999), "Multi-Agent Cooperation, Dynamic Workflow and XML for E-Commerce Automation", HP Labs.

Çingil, İ., (2000), "MESCHAIN: An Architecture for Supply Chain Integration and Automation on the Internet", Doktora Tezi, Ortadoğu Üniversitesi.

Dills, J., (1999), "XML: A Foundation for Intelligent Business Registries", Research Bulletin #99-02, 12 Haziran 1999.

Doğaç, A., (2001), "POEM: A Platform for an Open Electronic Marketplace Based on eCo Framework", working paper.

Doğaç, A., Arpınar, S., (2000), "Provision of Market Services for eCo Compliant Electronic Marketplaces", in ACM Sigmod Record, Vol.29, No.3.

Doğaç, A., Durusoy, İ., Arpınar, S., Tatbul, N., Köksal, P., (1999), "An Electronic Marketplace Architecture", in Current Trends in Data Management Technology, Idea Group Publishing.

Dolanbay, C., (2000), E-ticaret Strateji ve Yöntemler, Meteksan Sistem A.Ş., Ankara.

Graham, G., Hardaker, G., (2000), "Supply-Chain Management across the Internet", International Journal of Physical Distribution&Logistics Management, Vol. 30 no.3/4, MCB University Press, 286-295.

Griss, M., (2000), "My Agent Will Call Your Agent... But Will It Respond?", Software Development Magazine.

Kurnaz, S., Kilimci, P., (1999), "Elektronik Ticarete Kullanılan Bilgi Teknolojileri", Marmara Üniversitesi İstatistik ve Ekonometri Araştırma ve Uygulama Merkezi Dergisi, Yıl 1999, Sayı 1.

Microsoft BizTalk Server, (2000), "BizTalk Framework 2.0: Document and Message Specification".

RosettaNet, (2000), "Clusters, Segments and PIPs".

RosettaNet, (1998), "Partner Interface Process Technical Architecture".

Utkun, G., (1998), "An Agent Architecture for an Electronic Marketplace", Yüksek Lisans Tezi, Ortadoğu Üniversitesi.

Vogel, A., Duddy, K., (1997), Java Programming with CORBA, John Wiley & Sons Inc., New York.

Yilmaz, M., (1999), "Design and Implementation of Agent Architecture for an Electronic Marketplace", Yüksek Lisans Tezi, Ortadoğu Üniversitesi.

## İNTERNET KAYNAKLARI

1. CBL, "Common Business Library", <http://www.commerceone.com/xml/cbl/index.html>.
2. CommerceNet, (1997), "Catalogs for the Digital Marketplace", Research Report Note#97-03, <http://www.commerce.net/research/reports/199703.html>.
3. Doğaç, A., (1998), "A Survey of the Current State-of-the-Art in Electronic Commerce and Research Issues in Enabling Technologies", Euro-med Net 98 Conference, Electronic Commerce Track, Mart 1998, <http://www.srdc.metu.edu.tr/publications.html#2000>.
4. DOM, (1998), "Document Object Model Level 1 Specification", W3C Recommendation, <http://www.w3.org/TR/REC-DOM-Level-1>.
5. eCo Framework Project, (1999), "eCo Architecture for Electronic Commerce Interoperability", <http://www.commerce.net>.
6. eCo Framework Project, (1999), "eCo Semantic Recommendation", <http://www.commerce.net>.
7. Ersoy, Z., "Elektronik Ticaret ve Ticaret Noktaları", IGEME Yayını, <http://www.igeme.org.tr/TUR/etrade/eticaret/et1.htm>.
8. Graham, G., (2001), "Applying Web Services to Applications", <http://www-106.ibm.com/developerworks/webservices/library/ws-peer1.html?dwzone=ws>.
9. Hermans, B., (1997), "Intelligent Software Agents On The Internet", First Monday, Vol 2, No 3, 3 Mart 1997, [http://www.firstmonday.dk/issues/issue2\\_3/ch\\_123/index.html](http://www.firstmonday.dk/issues/issue2_3/ch_123/index.html).
10. Labrou, Y., Finin, T., (1994), "A Semantics Approach for KQML-A General Purpose Communication Language for Software Agents", 3ncü Uluslararası Information & Knowledge Konferansı, Kasım 1994, <http://www.cs.umbc.edu/kqml/papers/kqml-semantics.ps>.

11. Vawter, C., Roman, E., (2001), "J2EE vs. Microsoft.NET", The Middleware Company, <http://www.middleware-company.com/>.
12. XML, (1998), "Namespaces in XML", <http://www.w3.org/tr/1998/NOTE-xml-names>.
13. XSL, (2000), "Extensible Stylesheet Language (XSL) Version 1.0", W3C Working Draft, <http://www.w3.org/TR/xsl>.
14. XSLT, (1999), "(XSL) Transformations(XSLT) Version 1.0", W3C Recommendation, <http://www.w3.org/TR/xslt>.



**EKLER**

Ek 1	HKK Veritabanında Kullanılan Tablolar .....	138
Ek 2	Satıcı1 Veritabanında Kullanılan Tablolar.....	139
Ek 3	DTD Tanımları.....	140
Ek 4	Hava Kuvvetleri Komutanlığı için ASP Kodları .....	147
Ek 5	Hava Kuvvetleri Komutanlığı için COM Kodları.....	197
Ek 6	Satıcı1 için ASP Kodları .....	213
Ek 7	Satıcı1 için COM Kodları .....	227
Ek 8	Windows Script Components.....	248
Ek 9	Organizasyon Tanımları.....	251
Ek 10	Dosya Alım Fonksiyonu Tanımları.....	252
Ek 11	Port Tanımları .....	253
Ek 12	Kuyruk Tanımları.....	255
Ek 13	ODBC Tanımları.....	256
Ek 14	Virtual Directory Tanımları .....	257
Ek 15	Elektronik Ticaret Terimler Sözlüğü (Bozkurt, 2000).....	258



**Ek 1 HKK Veritabanında Kullanılan Tablolar****T\_AIEmNumara Tablosu**

AIEmNo	int
--------	-----

**T\_AIEmri Tablosu**

AIEmNo	int (unique index)
CalisanNo	int
Tarih	char(10)
KacKalemMal	int

**T\_AIEmriAyrinti Tablosu**

AIEmNo	int (AIEmNo + UniversalKod + SatirNo = unique index)
UniversalKod	int
SatirNo	int
Adet	int

**T\_Calisan Tablosu**

CalisanNo	int (unique index)
Ad	char(30)
Soyad	char(30)

**T\_GelenTeklifler Tablosu**

AIEmNo	int (unique index)
AIEmToplami	decimal
KazananFirma	int
S1Toplam	decimal
S2Toplam	decimal
S3Toplam	decimal

**T\_GlnTklfAyrinti Tablosu**

AIEmNo	int (unique index)
UniversalKod	int
SatirNo	int
Adet	int
S1Fiyat	decimal
S1HesaplanmisFiyat	decimal
S2Fiyat	decimal
S2HesaplanmisFiyat	decimal
S3Fiyat	decimal
S3HesaplanmisFiyat	decimal

**T\_Kurum Tablosu**

KurumNo	int (unique index)
Ad	char(30)

**T\_Mal Tablosu**

MalNo	int
UniversalKod	int (unique index)
Tanim	char(30)
Br	char(2)

**Ek 2 Satıcı1 Veritabanında Kullanılan Tablolar****T\_Kurum Tablosu**

KurumNo	int (unique index)
Ad	char(30)

**T\_Mal Tablosu**

MalNo	int
UniversalKod	int (unique index)
Tanim	char(30)
Br	char(2)
BrFiyat	decimal

**T\_Teklif Tablosu**

TeklifNo	int (unique index)
ALemNo	int
CalisanNo	int
Calisan	char(30)
ALemTarih	char(10)
KacKalemMal	int
ALemToplami	decimal
TeklifIsteyenFirma	int
Siparis	char(1)

**T\_TeklifAyrinti Tablosu**

TeklifNo	int (TeklifNo + UniversalKod + SatirNo = unique index)
UniversalKod	int
SatirNo	int
Adet	int
Fiyat	decimal
HesaplanmisFiyat	decimal

**T\_TeklifNumara Tablosu**

TeklifNo	int
----------	-----

### Ek 3 DTD Tanımları

#### *T\_ALEmIst.xml*

```

<?xml version="1.0"?>
<!-- Generated by using BizTalk Editor on Mon, Sep 25 2000 09:54:36 PM -->
<!-- Microsoft Corporation (c) 2000 (http://www.microsoft.com) -->
<Schema name="ALEmIst" b: BizTalkServerEditorTool_Version="1.0"
b:root_reference="ALEmIst" b:standard="XML" xmlns="urn:schemas-microsoft-com:xml-
data" xmlns:b="urn:schemas-microsoft-com: BizTalkServer" xmlns:d="urn:schemas-
microsoft-com:datatypes">
<b: SelectionFields/>

<ElementType name="Toplam" content="empty" model="closed">
<b: RecordInfo/>
<AttributeType name="KacKalemMal" d:type="int">
<b: FieldInfo/></AttributeType>
<attribute type="KacKalemMal" required="no"/>
</ElementType><ElementType name="Mal" content="empty" model="closed">
<b: RecordInfo/>
<AttributeType name="UniversalKod" d:type="int">
<b: FieldInfo/></AttributeType>
<AttributeType name="Tanim" d:type="string" d:maxLength="30" d:minLength="1">
<b: FieldInfo/></AttributeType>
<AttributeType name="SatirNo" d:type="int">
<b: FieldInfo/></AttributeType>
<AttributeType name="OlcuBirimi" d:type="string" d:maxLength="2" d:minLength="1">
<b: FieldInfo/></AttributeType>
<AttributeType name="Adet" d:type="int">
<b: FieldInfo/></AttributeType>
<attribute type="Tanim" required="no"/>
<attribute type="UniversalKod" required="no"/>
<attribute type="OlcuBirimi" required="no"/>
<attribute type="Adet" required="no"/>
<attribute type="SatirNo" required="no"/>
</ElementType><ElementType name="CalisanBilgisi" content="empty" model="closed">
<b: RecordInfo/>
<AttributeType name="Soyad" d:type="string" d:maxLength="30" d:minLength="1">
<b: FieldInfo/></AttributeType>
<AttributeType name="CalisanNo" d:type="int">
<b: FieldInfo/></AttributeType>
<AttributeType name="Ad" d:type="string" d:maxLength="30" d:minLength="1">
<b: FieldInfo/></AttributeType>
<attribute type="Soyad" required="no"/>
<attribute type="Ad" required="no"/>
<attribute type="CalisanNo" required="no"/>
</ElementType><ElementType name="Baslik" content="empty" model="closed">
<b: RecordInfo/>
<AttributeType name="Tarih" d:type="date">
<b: FieldInfo/></AttributeType>
<AttributeType name="KurumNo" d:type="int">

```



```
<b:FieldInfo/></AttributeType>
<AttributeType name="AlEmNo" d:type="int">
<b:FieldInfo/></AttributeType>
<attribute type="Tarih" required="no"/>
<attribute type="AlEmNo" required="no"/>
<attribute type="KurumNo" required="no"/>
</ElementType><ElementType name="AlEmIst" content="eltOnly" model="open">
<b:RecordInfo/>
<element type="Baslik" maxOccurs="1" minOccurs="1"/>
<element type="CalisanBilgisi" maxOccurs="1" minOccurs="1"/>
<element type="Mal" maxOccurs="*" minOccurs="1"/>
<element type="Toplam" maxOccurs="1" minOccurs="1"/>
</ElementType></Schema>
```



*T\_ALEm.xml*

```

<?xml version="1.0"?>
<!-- Generated by using BizTalk Editor on Tue, Sep 26 2000 11:47:58 PM -->
<!-- Microsoft Corporation (c) 2000 (http://www.microsoft.com) -->
<Schema name="ALEm" b: BizTalkServerEditorTool_Version="1.0"
b:root_reference="ALEm" b:standard="XML" xmlns="urn:schemas-microsoft-com:xml-data"
xmlns:b="urn:schemas-microsoft-com: BizTalkServer" xmlns:d="urn:schemas-microsoft-
com:datatypes">
<b: SelectionFields/>

<ElementType name="Toplam" content="empty" model="closed">
<b: RecordInfo/>
<AttributeType name="KacKalemMal" d:type="int">
<b: FieldInfo/></AttributeType>
<AttributeType name="ALEmToplami" d:type="r8">
<b: FieldInfo/></AttributeType>
<attribute type="KacKalemMal" required="no"/>
<attribute type="ALEmToplami" required="no" default="0"/>
</ElementType><ElementType name="Mal" content="empty" model="closed">
<b: RecordInfo/>
<AttributeType name="UniversalKod" d:type="int">
<b: FieldInfo/></AttributeType>
<AttributeType name="Tanim" d:type="string" d:maxLength="30" d:minLength="1">
<b: FieldInfo/></AttributeType>
<AttributeType name="SatirNo" d:type="int">
<b: FieldInfo/></AttributeType>
<AttributeType name="OlcuBirimi" d:type="string" d:maxLength="2" d:minLength="1">
<b: FieldInfo/></AttributeType>
<AttributeType name="HesaplanmisFiyat" d:type="r8">
<b: FieldInfo/></AttributeType>
<AttributeType name="Fiyat" d:type="r4">
<b: FieldInfo/></AttributeType>
<AttributeType name="Adet" d:type="int">
<b: FieldInfo/></AttributeType>
<attribute type="Tanim" required="no"/>
<attribute type="UniversalKod" required="no"/>
<attribute type="OlcuBirimi" required="no"/>
<attribute type="Adet" required="no"/>
<attribute type="Fiyat" required="no" default="0"/>
<attribute type="HesaplanmisFiyat" required="no" default="0"/>
<attribute type="SatirNo" required="no"/>
</ElementType><ElementType name="CalisanBilgisi" content="empty" model="closed">
<b: RecordInfo/>
<AttributeType name="Soyad" d:type="string" d:maxLength="30" d:minLength="1">
<b: FieldInfo/></AttributeType>
<AttributeType name="CalisanNo" d:type="int">
<b: FieldInfo/></AttributeType>
<AttributeType name="Ad" d:type="string" d:maxLength="30" d:minLength="1">
<b: FieldInfo/></AttributeType>
<attribute type="Soyad" required="no"/>

```

```
<attribute type="Ad" required="no"/>
<attribute type="CalisanNo" required="no"/>
</ElementType><ElementType name="Baslik" content="empty" model="closed">
<b:RecordInfo/>
<AttributeType name="Tarih" d:type="date">
<b:FieldInfo/></AttributeType>
<AttributeType name="KurumNo" d:type="int">
<b:FieldInfo/></AttributeType>
<AttributeType name="AlEmNo" d:type="int">
<b:FieldInfo/></AttributeType>
<attribute type="Tarih" required="no"/>
<attribute type="AlEmNo" required="no"/>
<attribute type="KurumNo" required="no"/>
</ElementType><ElementType name="AlEm" content="eltOnly" model="open">
<b:RecordInfo/>
<element type="Baslik" maxOccurs="1" minOccurs="1"/>
<element type="CalisanBilgisi" maxOccurs="1" minOccurs="1"/>
<element type="Mal" maxOccurs="*" minOccurs="1"/>
<element type="Toplam" maxOccurs="1" minOccurs="1"/>
</ElementType></Schema>
```



***T\_Fatura.xml***

```

<?xml version="1.0"?>
<!-- Generated by using BizTalk Editor on Sun, Aug 27 2000 07:20:18 AM -->
<!-- Microsoft Corporation (c) 2000 (http://www.microsoft.com) -->
<Schema name="Fatura" b: BizTalkServerEditorTool_Version="1.0"
b:root_reference="Fatura" b:standard="XML" xmlns="urn:schemas-microsoft-com:xml-data"
xmlns:b="urn:schemas-microsoft-com: BizTalkServer" xmlns:d="urn:schemas-microsoft-
com:datatypes">
<b:SelectionFields/>

<ElementType name="Toplam" content="empty" model="closed">
<b:RecordInfo/>
<AttributeType name="VergiMiktari" d:type="r8">
<b:FieldInfo/></AttributeType>
<AttributeType name="KacKalemMal" d:type="int">
<b:FieldInfo/></AttributeType>
<AttributeType name="IndirimMiktari" d:type="r8">
<b:FieldInfo/></AttributeType>
<AttributeType name="FaturaToplami" d:type="r8">
<b:FieldInfo/></AttributeType>
<AttributeType name="AlEmToplami" d:type="r8">
<b:FieldInfo/></AttributeType>
<attribute type="KacKalemMal" required="no"/>
<attribute type="AlEmToplami" required="no" default="0"/>
<attribute type="IndirimMiktari" required="no" default="0"/>
<attribute type="VergiMiktari" required="no"/>
<attribute type="FaturaToplami" required="no"/>
</ElementType><ElementType name="Satici" content="eltOnly" model="closed">
<b:RecordInfo/>
<element type="Adres" maxOccurs="1" minOccurs="0"/>
<element type="KontaktBilgisi" maxOccurs="1" minOccurs="0"/>
</ElementType><ElementType name="Mal" content="empty" model="closed">
<b:RecordInfo/>
<AttributeType name="UniversalKod" d:type="int">
<b:FieldInfo/></AttributeType>
<AttributeType name="Tanim" d:type="string" d:maxLength="30" d:minLength="1">
<b:FieldInfo/></AttributeType>
<AttributeType name="SatirNo" d:type="int">
<b:FieldInfo/></AttributeType>
<AttributeType name="OlcuBirimi" d:type="string" d:maxLength="2" d:minLength="1">
<b:FieldInfo/></AttributeType>
<AttributeType name="HesaplanmisFiyat" d:type="r8">
<b:FieldInfo/></AttributeType>
<AttributeType name="Fiyat" d:type="r4">
<b:FieldInfo/></AttributeType>
<AttributeType name="Adet" d:type="int">
<b:FieldInfo/></AttributeType>
<attribute type="Tanim" required="no"/>
<attribute type="UniversalKod" required="no"/>
<attribute type="OlcuBirimi" required="no"/>

```

```

<attribute type="Adet" required="no"/>
<attribute type="Fiyat" required="no" default="0"/>
<attribute type="HesaplanmisFiyat" required="no" default="0"/>
<attribute type="SatirNo" required="no"/>
</ElementType><ElementType name="KontaktBilgisi" content="empty" model="closed">
<b:RecordInfo/>
<AttributeType name="CalisanNo" d:type="int">
<b:FieldInfo/></AttributeType>
<AttributeType name="Ad" d:type="string" d:maxLength="60" d:minLength="1">
<b:FieldInfo/></AttributeType>
<attribute type="Ad" required="no"/>
<attribute type="CalisanNo" required="no"/>
</ElementType><ElementType name="GondermeAdresi" content="eltOnly"
model="closed">
<b:RecordInfo/>
<element type="Adres" maxOccurs="1" minOccurs="0"/>
</ElementType><ElementType name="FaturaSatisSartlari" content="empty"
model="closed">
<b:RecordInfo/>
<AttributeType name="IndirimOrani" d:type="r4">
<b:FieldInfo/></AttributeType>
<attribute type="IndirimOrani" required="no"/>
</ElementType><ElementType name="FaturaAdresi" content="eltOnly" model="closed">
<b:RecordInfo/>
<element type="Adres" maxOccurs="1" minOccurs="0"/>
</ElementType><ElementType name="Fatura" content="eltOnly" model="open">
<b:RecordInfo/>
<element type="Baslik" maxOccurs="1" minOccurs="1"/>
<element type="Satici" maxOccurs="1" minOccurs="0"/>
<element type="Alici" maxOccurs="1" minOccurs="0"/>
<element type="FaturaSatisSartlari" maxOccurs="1" minOccurs="0"/>
<element type="Mal" maxOccurs="*" minOccurs="1"/>
<element type="Toplam" maxOccurs="1" minOccurs="1"/>
<element type="FaturaAdresi" maxOccurs="1" minOccurs="0"/>
<element type="GondermeAdresi" maxOccurs="1" minOccurs="0"/>
</ElementType><ElementType name="Baslik" content="empty" model="closed">
<b:RecordInfo/>
<AttributeType name="Tarih" d:type="date">
<b:FieldInfo/></AttributeType>
<AttributeType name="Numara" d:type="string">
<b:FieldInfo/></AttributeType>
<AttributeType name="GondermeTarihi" d:type="date">
<b:FieldInfo/></AttributeType>
<AttributeType name="AlEmTarih" d:type="date">
<b:FieldInfo/></AttributeType>
<AttributeType name="AlEmNo" d:type="int">
<b:FieldInfo/></AttributeType>
<attribute type="Tarih" required="no"/>
<attribute type="Numara" required="no" default="0"/>
<attribute type="AlEmNo" required="no"/>
<attribute type="AlEmTarih" required="no"/>

```

```
<attribute type="GondermeTarihi" required="no"/>
</ElementType><ElementType name="Alici" content="eltOnly" model="closed">
<b:RecordInfo/>
<element type="Adres" maxOccurs="1" minOccurs="0"/>
<element type="KontaktBilgisi" maxOccurs="1" minOccurs="0"/>
</ElementType><ElementType name="Adres" content="empty" model="closed">
<b:RecordInfo/>
<AttributeType name="PostaKodu" d:type="string" d:maxLength="15" d:minLength="1">
<b:FieldInfo/></AttributeType>
<AttributeType name="II" d:type="string" d:maxLength="30" d:minLength="1">
<b:FieldInfo/></AttributeType>
<AttributeType name="Adres2" d:type="string" d:maxLength="55" d:minLength="1">
<b:FieldInfo/></AttributeType>
<AttributeType name="Adres1" d:type="string" d:maxLength="55" d:minLength="1">
<b:FieldInfo/></AttributeType>
<AttributeType name="Ad" d:type="string" d:maxLength="60" d:minLength="1">
<b:FieldInfo/></AttributeType>
<attribute type="Ad" required="no"/>
<attribute type="Adres1" required="no"/>
<attribute type="Adres2" required="no"/>
<attribute type="II" required="no"/>
<attribute type="PostaKodu" required="no"/>
</ElementType></Schema>
```

#### Ek 4 Hava Kuvvetleri Komutanlığı için ASP Kodları

##### Calisan.asp

```

<%@ Language=VBScript.%>
<HTML>
  <HEAD> <TITLE>HKK - Çal&#305;&#351;an Bilgileri </TITLE>
  <STYLE>
    body {font-family:verdana,helvetica;font-size:8pt}
    button {background-color:"#001964";font-
family:verdana,helvetica;color="WHITE";font-size:8pt;width:70}
    input {font-family:verdana,helvetica;font-size:8pt}
    Select {font-family:verdana,helvetica;font-size:8pt; valign:top;}
    .Table {background:black}
    .TableHead { font:bold; color:white; background:black}
    .TableColumnHead { font:normal 'Verdana' bold; color:white;background:#6699CC}
    .TableRow {font:x-small 'Verdana'; color:black;background:#99CCFF}
  </STYLE>

</HEAD>

<BODY LANGUAGE=javascript onerror="return window_onerror()">

<!-- #Include File="VeritabaninaBaglan.ASP" -->
<FORM METHOD=post id=frmCalisan name=frmCalisan>

<table border="0" cellspacing="0" cellpadding="20" height="200" width="95%"
class="Table">
<tr class="TableRow">
  <td width="10%" bgcolor="WHITE">
    &nbsp;
  </td>
  <td bgcolor="LIGHTYELLOW">
<a href="default.asp"> HKK </a>
  </td>
  <td bgcolor="#99CCFF">
<a href="calisan.asp"> Çal&#305;&#351;an</a>
  </td>
  <td bgcolor="#904eff">
<a href="mal.asp"> Mal </a>
  </td>
  <td bgcolor="#ffff00">
<a href="kurum.asp"> Kurum </a>
  </td>
  <td bgcolor="#669900">
<a href="hkkalimEmri.asp">Al&#305;m Emri </a>
  </td>
  <td bgcolor="#CC0066">

```

```

<a href="teklif.asp"> Teklifler </a>
</td>

<td width="10%" bgcolor="WHITE">
  &nbsp;
</td>
</tr>
<tr class="TableRow">
<td colspan="8" height="175" bgcolor="#99CCFF" valign="TOP" >
<H1 align=center>Çalışan Personel Bilgileri</H1>
<font face="Arial" SIZE="2">

<%
Dim i, j

Client= Request.ServerVariables("HTTP_USER_AGENT")
ClientVer=Split(Client, ";")
rs.Open ("select * from t_calisan order by ad, soyad"), con, 1%>

<TABLE align=center class="Table">
  <tr >
    <TD VAlign=center> <select size='15' name='selCalisan' onchange=getNumara() >
      <% Do While Not rs.EOF%>

        <option value='<%=Trim(rs("CalisanNo")) %>'> <%= Trim(rs("Ad")) %> &nbsp;<%= Trim(rs("Soyad")) %></option>
        <% rs.MoveNext
      Loop
      rs.close %>
    </select></td></tr>
  <tr>
    <TD><INPUT type='hidden' NAME="txtCalisanNo" Id="txtCalisanNo" maxLength = "5"
size = "5"></TD>
  </TR>
</table>
<% con.close
  set rs=Nothing
  set con=Nothing %>
<table align=center border="0" bgcolor="#B0C4DE">
<tr>
<td><button id="addnew" onclick="Ekle()">Ekle</button></td>
<td><button id="duzelt" onclick="Duzelt()">Düzelt</button></td>
<td><button id="sil" onclick="Sil()">Sil</button></td>
</tr>
</table>

</td>
</tr>
</table>

```



```
<script language="Javascript">
```

```
var mywin
var prevdiv="dummydiv"
```

```
function getNumara()
{
var i;
i = frmCalisan.elements('selCalisan').value;
frmCalisan.elements('txtCalisanNo').value= i;
}
```

```
function Sil()
{
var str, i;
i=frmCalisan.elements('txtCalisanNo').value ;
str="silCalisan.asp?ComponentID="+i;
```


```
if (mywin!=null)
{
if (!mywin.closed)
{
mywin.close();
mywin=null;
}
}
if (i != "")
{
mywin=window.open(str,null,"directories=no,location=no,menubar=no,status,toolbar=no,resizeable,scrollbars");
window.event.cancelBubble=true;
}
}
```

```
function Ekle()
{
var str;
str="EkleCalisan.asp";
if (mywin!=null)
{
if (!mywin.closed)
{
mywin.close();
mywin=null;
}
}
mywin=window.open(str,null,"directories=no,location=no,menubar=no,status,toolbar=no,resizeable,scrollbars");
window.event.cancelBubble=true;
}
function Duzelt()
```

```
{
var str, i;
i=frmCalisan.elements('txtCalisanNo').value ;
str="duzeltCalisan.asp?ComponentID="+i;

if (mywin!=null)
{
if (!mywin.closed)
{
mywin.close();
mywin=null;
}
}
if (i != "")
{
mywin=window.open(str,null,"directories=no,location=no,menubar=no,status,toolbar=no,r
esizable,scrollbars");
window.event.cancelBubble=true;
}
}

</script>
</html>
```



**Default.asp**

```

<HTML>
  <HEAD> <TITLE>Hava Kuvvetleri Komutanl&#305;&#287;&#305;</TITLE>
</HEAD>
  <STYLE>
    body {font-family:verdana,helvetica;font-size:8pt}
    button {background-color:"#001964";font-
family:verdana,helvetica;color="WHITE";font-size:8pt;width:70}
    input {font-family:verdana,helvetica;font-size:8pt}
    Select {font-family:verdana,helvetica;font-size:8pt; valign:top;}
    .Table {background:black}
    .TableHead { font:bold; color:white; background:black}
    .TableColumnHead { font:normal 'Verdana' bold; color:white;background:#6699CC}
    .TableRow {font:x-small 'Verdana'; color:black;background:#99CCFF}
  </STYLE>
<body>

<table border="0" cellspacing="0" cellpadding="20" height="200" width="95%"
class="Table">
<tr class="TableRow">
  <td width="10%" bgcolor="WHITE">
    &nbsp;
  </td>
  <td bgcolor="LIGHTYELLOW">
    <a href="default.asp"> HKK </a>
  </td>
  <td bgcolor="#99CCFF">
    <a href="calisan.asp"> Çal&#305;&#351;an </a>
  </td>
  <td bgcolor="#904eff">
    <a href="mal.asp"> Mal </a>
  </td>
  <td bgcolor="#ffff00">
    <a href="kurum.asp"> Kurum </a>
  </td>
  <td bgcolor="#669900">
    <a href="hkkalimEmri.asp"> Al&#305;m Emri </a>
  </td>
  <td bgcolor="#CC0066">
    <a href="teklif.asp"> Teklifler </a>
  </td>

  <td width="10%" bgcolor="WHITE">
    &nbsp;
  </td>
</tr>
<tr class="TableRow">
<td colspan="8" height="175" bgcolor="LIGHTYELLOW" valign="TOP" >
<font face="Arial" SIZE="2">

```

<P><BR> Hava Kuvvetleri Komutanl&#305;&#287;&#305; sayfas&#305;na HO&#350;  
GELD&#304;N&#304;Z!

<BR> <BR><br><br>

Bu sayfa ile Çal&#305;&#351;an, Mal, Kurum ve Al&#305;m Emri veri giri&#351;lerini  
yapabilirsiniz.<BR> <BR>

Verilen teklifleri ve kazanan firmay&#305; görebilirsiniz.</P>

</font>

</td>

</tr>

</table>

</body>

</html>



**DuzeltCalisan.asp**

```

<HTML>
  <HEAD>
    <meta http-equiv="Content-Language" content="tr">
    <meta name="GENERATOR" content="Microsoft FrontPage 4.0">
    <meta name="ProgId" content="FrontPage.Editor.Document">
    <TITLE>HKK - Çal&#305;&#351;an Düzeltme </TITLE> <STYLE>
      body {font-family:verdana,helvetica;font-size:8pt}
      button {background-color:"#001964";font-
family:verdana,helvetica;color="WHITE";font-size:8pt;width:70}
      input {font-family:verdana,helvetica;font-size:8pt}
      Select {font-family:verdana,helvetica;font-size:8pt; valign:top;}
      .Table {background:black}
      .TableHead { font:bold; color:white; background:black}
      .TableColumnHead { font:normal 'Verdana' bold; color:white;background:#6699CC}
      .TableRow {font:x-small 'Verdana'; color:black;background:#99CCFF}
    </STYLE>

<%@ Language=VBScript %>
<!-- #Include File="VeritabaninaBaglan.ASP" -->

<FORM METHOD="post" id=frmDuzelt name=frmDuzelt>

</HEAD>

<BODY bgcolor="#99CCFF">
<input type="hidden" name="hNo" size="5" value =
"<%=Request.QueryString("ComponentID")%>" >
<%

Dim calNo

calNo=Request.QueryString("ComponentID")
Client= Request.ServerVariables("HTTP_USER_AGENT")
ClientVer=Split(Client, ";")
rs.Open ("select * from t_calisan where calisanNo = " & calNo ), con, 1

%>
<center>
<table border="0">
<tr>
  <td width=60>Ad :</td>
  <td><input type="text" name="txtAd" size="30" value= <%= trim(rs("Ad"))%> ></td>
</tr>
<tr>
  <td width=60>Soyad :</td>

```

```

    <td><input type="text" name="txtSoyad" size="30" value= <%= trim(rs("soyAd"))%> >
</td>
</tr>

```

```

<tr>
<td colspan=2 align=center>
<button onclick="Duzelt()" >Kaydet</button>

<button onclick="window.close()">Kapat</button>
</td>
</tr>
</table>

```

```

</form>
</center>
<% rs.close
con.close
set rs=Nothing
set con=Nothing %>
</BODY>

```

```

<script language="Javascript">
var mywin
var prevdiv="dummydiv"

```

```
function Duzelt()
```

```
{
var str, i, j, k;
```

```

i=frmDuzelt.elements('hNo').value ;
j=frmDuzelt.elements('txtAd').value ;
k=frmDuzelt.elements('txtSoyad').value ;

```

```
str="duzeltCalisan2.asp?ID="+ i + "&ad=" + j + "&soyad=" + k;
```

```

if (mywin!=null)
{
if (!mywin.closed)
{
mywin.close();
mywin=null;
}
}

```

```
mywin=window.open(str,null,"directories=no,location=no,menubar=no,status,toolbar=no,resizeable,scrollbars");
```

```
window.event.cancelBubble=true;
```

```
}
```

```

</script>
</HTML>

```

**DuzeltCalisan2.asp**

```

<HTML>
  <HEAD> <TITLE>HKK - Çal&#305;&#351;an Düzeltme </TITLE>
  <STYLE>
    body {font-family:verdana,helvetica;font-size:8pt}
    button {background-color:"#001964";font-
family:verdana,helvetica;color="WHITE";font-size:8pt;width:70}
    input {font-family:verdana,helvetica;font-size:8pt}
    Select {font-family:verdana,helvetica;font-size:8pt; valign:top;}
    .Table {background:black}
    .TableHead { font:bold; color:white; background:black}
    .TableColumnHead { font:normal 'Verdana' bold; color:white;background:#6699CC}
    .TableRow {font:x-small 'Verdana'; color:black;background:#99CCFF}
  </STYLE>

<%@ Language=VBScript %>

<!-- #Include File="VeritabaninaBaglan.ASP" -->
</HEAD>
<BODY bgcolor="#99CCFF">
<%
  Dim ad, soyad, calNo
  Client= Request.ServerVariables("HTTP_USER_AGENT")
  ClientVer=Split(Client,",")
  calNo=Request.QueryString("ID")
  ad=Request.QueryString("ad")
  soyad=Request.QueryString("soyad")

  rs.Open ("select * from t_calisan where ad="" & ad & "" and soyad="" & soyad &"""), con, 1
  if (rs.eof) then
    con.Execute "UPDATE t_Calisan SET ad = "" & ad & "" , soyad = "" & soyad & ""
  WHERE calisanNo = " & calNo ,adCmdText + adExecuteNoRecords
    response.write calNo & " numaral&#305; kay&#305;t " & ad & " " & soyad & " bilgisi ile
  güncelle&#351;tirildi..."
    else
    response.write ad & " " & soyad & " bilgisi veritaban&#305;nda var. Farkl&#305; bir
  bilgi ile güncelle&#351;tiriniz...."
  end if

  rs.close
  con.close
  set rs=Nothing
  set con=Nothing %>
<center><br>
<table border="0">
<tr><td colspan=2 align=center>
<button onclick="window.close()">Kapat</button>
</td></tr></table></center>
</body>
</html>

```

**DuzeltKurum.asp**

```

<HTML>
  <HEAD> <TITLE>HKK - Kurum Düzeltme</TITLE>
  <STYLE>
    body {font-family:verdana,helvetica;font-size:8pt}
    button {background-color:"#001964";font-
family:verdana,helvetica;color="WHITE";font-size:8pt;width:70}
    input {font-family:verdana,helvetica;font-size:8pt}
    Select {font-family:verdana,helvetica;font-size:8pt; valign:top;}
    .Table {background:black}
    .TableHead { font:bold; color:white; background:black}
    .TableColumnHead { font:normal 'Verdana' bold; color:white;background:#6699CC}
    .TableRow {font:x-small 'Verdana'; color:black;background:#99CCFF}
  </STYLE>

  <%@ Language=VBScript %>
  <!-- #Include File="VeritabaninaBaglan.ASP" -->

  <FORM METHOD="post" id=frmDuzelt name=frmDuzelt>

  </HEAD>

  <BODY bgcolor="#ffff00">
  <input type="hidden" name="kNo" size="5" value =
  "<%=Request.QueryString("ComponentID")%>" >
  <%

  Dim kurumNo

  kurumNo=Request.QueryString("ComponentID")
  Client= Request.ServerVariables("HTTP_USER_AGENT")
  ClientVer=Split(Client,",")
  rs.Open ("select * from t_kurum where kurumNo = " & kurumNo ), con, 1

  %>

  <center>
  <table border="0">

  <tr>
    <td width=60>Kurum :</td>
    <td><input type="text" name="txtKurum" size="30" value= <%= trim(rs("ad"))%> ></td>
  </tr>
  <tr>

```



```

<td colspan=2 align=center>
<button onclick="Duzelt()">Kaydet</button>

<button onclick="window.close()">Kapat</button>
</td>
</tr>
</table>

</form>
</center>
<% rs.close
con.close
set rs=Nothing
set con=Nothing %>
</BODY>

<script language="Javascript">
var mywin
var prevdiv="dummydiv"

function Duzelt()
{
var str, i, j;

i=frmDuzelt.elements('kNo').value ;
j=frmDuzelt.elements('txtKurum').value ;

str="duzeltKurum2.asp?ID="+ i + "&kurum=" + j;

if (mywin!=null)
{
if (!mywin.closed)
{
mywin.close();
mywin=null;
}
}
mywin=window.open(str,null,"directories=no,location=no,menubar=no,status,toolbar=no,r
esizable,scrollbars");
window.event.cancelBubble=true;
}

</script>
</HTML>

```

**DuzeltKurum2.asp**

```

<HTML>
  <HEAD> <TITLE>HKK - Kurum Düzeltme </TITLE>
  <STYLE>
    body {font-family:verdana,helvetica;font-size:8pt}
    button {background-color:"#001964";font-
family:verdana,helvetica;color="WHITE";font-size:8pt;width:70}
    input {font-family:verdana,helvetica;font-size:8pt}
    Select {font-family:verdana,helvetica;font-size:8pt; valign:top;}
    .Table {background:black}
    .TableHead { font:bold; color:white; background:black}
    .TableColumnHead { font:normal 'Verdana' bold; color:white;background:#6699CC}
    .TableRow {font:x-small 'Verdana'; color:black;background:#99CCFF}
  </STYLE>

  <%@ Language=VBScript %>
  <!-- #Include File="VeritabaninaBaglan.ASP" -->

  </HEAD>

  <BODY bgcolor="#ffff00">
  <%

Dim kurum,kurumNo
Client= Request.ServerVariables("HTTP_USER_AGENT")
ClientVer=Split(Client, ";")
kurumNo=Request.QueryString("ID")
kurum=Request.QueryString("kurum")

rs.Open ("select * from t_kurum where ad='" & kurum & """), con, 1
if (rs.eof) then
  con.Execute "UPDATE t_kurum SET ad = '" & kurum & "' WHERE kurumNo = " &
kurumNo ,adCmdText + adExecuteNoRecords
  response.write kurumNo & " numaral&#305; kay&#305;t " & kurum & " bilgisi ile
güncelle&#351;tirildi...."
  else
  response.write kurum & " bilgisi veritaban&#305;nda var. Farkl&#305; bir bilgi ile
güncelle&#351;tiriniz...."
  end if
rs.close
con.close
set rs=Nothing
set con=Nothing %>

<center><br>
<table border="0">
<tr><td colspan=2 align=center>
<button onclick="window.close()">Kapat</button>
</td></tr></table></center>
</body></html>

```

**DuzeltMal.asp**

```

<HTML>
  <HEAD> <TITLE>HKK - Mal Düzeltme </TITLE>
  <STYLE>
    body {font-family:verdana,helvetica;font-size:8pt}
    button {background-color:"#001964";font-
family:verdana,helvetica;color="WHITE";font-size:8pt;width:70}
    input {font-family:verdana,helvetica;font-size:8pt}
    Select {font-family:verdana,helvetica;font-size:8pt; valign:top;}
    .Table {background:black}
    .TableHead { font:bold; color:white; background:black}
    .TableColumnHead { font:normal 'Verdana' bold; color:white;background:#6699CC}
    .TableRow {font:x-small 'Verdana'; color:black;background:#99CCFF}
  </STYLE>

  <%@ Language=VBScript %>
  <!-- #Include File="VeritabaninaBaglan.ASP" -->

  <FORM METHOD="post" id=frmDuzelt name=frmDuzelt>

  </HEAD>

  <BODY bgcolor="#904eff">
  <input type="hidden" name="hNo" size="5" value =
  "<%=Request.QueryString("ComponentID")%>" >
  <%

  Dim malNo

  malNo=Request.QueryString("ComponentID")
  Client= Request.ServerVariables("HTTP_USER_AGENT")
  ClientVer=Split(Client,",")
  rs.Open ("select * from t_mal where malNo = " & malNo ), con, 1

  %>

  <center>
  <table border="0">

  <tr>
    <td width=60>UniversalKod :</td>
    <td><input type="text" name="txtUniversalKod" size="5" value= <%=
  trim(rs("universalKod"))%> ></td>
  </tr>
  <tr>
    <td width=60>Tan&#305;m :</td>
    <td><input type="text" name="txtTanim" size="30" value= <%= trim(rs("tanim"))%>
  ></td>
  </tr>

```

```

<tr>
  <td width=60>Birim :</td>
  <td><input type="text" name="txtBr" size="2" value= <%= trim(rs("br"))%> </td>
</tr>

<tr>
<td colspan=2 align=center>
<button onclick="Duzelt()">Kaydet</button>

<button onclick="window.close()">Kapat</button>

</td>
</tr>
</table>
<% rs.close
con.close
set rs=Nothing
set con=Nothing %>
</form>
</center>

</BODY>

<script language="Javascript">
var mywin
var prevdiv="dummydiv"
function Duzelt()
{
var str, i, j, k, t;

i=frmDuzelt.elements('hNo').value ;
j=frmDuzelt.elements('txtUniversalKod').value ;
k=frmDuzelt.elements('txtTanim').value ;
t=frmDuzelt.elements('txtBr').value ;

str="duzeltMal2.asp?ID="+ i + "&kod=" + j + "&tanim=" + k+ "&br=" + t;

if (mywin!=null)
{
if (!mywin.closed)
{
mywin.close();
mywin=null;
}
}
mywin=window.open(str,null,"directories=no,location=no,menubar=no,status,toolbar=no,r
esizable,scrollbars");
window.event.cancelBubble=true;
}
</script>
</HTML>

```

*DuzeltMal2.asp*

```

<HTML>
  <HEAD> <TITLE>HKK - Mal Düzeltme </TITLE>
  <STYLE>
    body {font-family:verdana,helvetica;font-size:8pt}
    button {background-color:"#001964";font-
family:verdana,helvetica;color="WHITE";font-size:8pt;width:70}
    input {font-family:verdana,helvetica;font-size:8pt}
    Select {font-family:verdana,helvetica;font-size:8pt; valign:top;}
    .Table {background:black}
    .TableHead { font:bold; color:white; background:black}
    .TableColumnHead { font:normal 'Verdana' bold; color:white;background:#6699CC}
    .TableRow {font:x-small 'Verdana'; color:black;background:#99CCFF}
  </STYLE>

<%@ Language=VBScript %>
<!-- #Include File="VeritabaninaBaglan.ASP" -->
</HEAD>
<BODY bgcolor="#904eff">
<%

Dim kod, tanim, br, malNo
Client= Request.ServerVariables("HTTP_USER_AGENT")
ClientVer=Split(Client,",")
malNo=Request.QueryString("ID")
kod=Request.QueryString("kod")
tanim=Request.QueryString("tanim")
br=Request.QueryString("br")

rs.Open ("select * from t_mal where universalKod="" & kod & """), con, 1
if (rs.eof) then
  con.Execute "UPDATE t_mal SET universalKod = "" & kod & "" , tanim = "" & tanim &
"" , br = "" & br & "" WHERE malNo = "" & malNo ,adCmdText + adExecuteNoRecords
  response.write malNo & " numaral&#305; kay&#305;t " & kod & " " & tanim & " bilgisi
ile güncelle&#351;tirildi...."
  else
    response.write kod & " " & tanim & " bilgisi veritaban&#305;nda var. Farkl&#305; bir
bilgi ile güncelle&#351;tiriniz...."
  end if
rs.close
con.close
set rs=Nothing
set con=Nothing %>

<center><br>
<table border="0">
<tr><td colspan=2 align=center>
<button onclick="window.close()">Kapat</button>
</td></tr>
</table></center></body></html>

```

**EkleCalisan.asp**

```

<html>
  <HEAD>
    <meta http-equiv="Content-Language" content="tr">
    <meta name="GENERATOR" content="Microsoft FrontPage 4.0">
    <meta name="ProgId" content="FrontPage.Editor.Document">
    <TITLE>HKK - Çal&#305;&#351;an Ekleme </TITLE> <STYLE>
      body {font-family:verdana,helvetica;font-size:8pt}
      button {background-color:"#001964";font-
family:verdana,helvetica;color="WHITE";font-size:8pt;width:70}
      input {font-family:verdana,helvetica;font-size:8pt}
      Select {font-family:verdana,helvetica;font-size:8pt; valign:top;}
      .Table {background:black}
      .TableHead { font:bold; color:white; background:black}
      .TableColumnHead { font:normal 'Verdana' bold; color:white;background:#6699CC}
      .TableRow {font:x-small 'Verdana'; color:black;background:#99CCFF}
    </STYLE>

<%@ Language=VBScript %>
<!-- #Include File="VeritabaninaBaglan.ASP" -->

<FORM METHOD="post" id=frmEkle name=frmEkle>
  </HEAD>
  <BODY bgcolor="#99CCFF">

  <center>
  <table border="0">
  <tr>
    <td width=60>Ad :</td>
    <td><input type="text" name="txtAd" size="30" ></td>
  </tr>
  <tr>
    <td width=60>Soyad :</td>
    <td><input type="text" name="txtSoyad" size="30" </td>
  </tr>

  <tr>
  <td colspan=2 align=center>
  <button onclick="Ekle()" >Kaydet</button>

  <button onclick="window.close()">Kapat</button>
  </td>
  </tr>

  </table>

</form>

```

```
</center>
</BODY>

<script language="Javascript">
var mywin
var prevdiv="dummydiv"

function Ekle()
{
var str, j, k;

j=frmEkle.elements('txtAd').value ;
k=frmEkle.elements('txtSoyad').value ;
if (j!=null)
{ str="ekleCalisan2.asp?ad=" + j + "&soyad=" + k; }

if (mywin!=null)
{
if (!mywin.closed)
{
mywin.close();
mywin=null;
}
}
mywin=window.open(str,null,"directories=no,location=no,menubar=no,status,toolbar=no,resizeable,scrollbars");
window.event.cancelBubble=true;
}

</script>
</HTML>
```

*EkleCalisan2.asp*

```

<HTML>
  <HEAD> <TITLE>HKK - Çal&#305;&#351;an Ekleme </TITLE>
  <STYLE>
    body {font-family:verdana,helvetica;font-size:8pt}
    button {background-color:"#001964";font-
family:verdana,helvetica;color="WHITE";font-size:8pt;width:70}
    input {font-family:verdana,helvetica;font-size:8pt}
    Select {font-family:verdana,helvetica;font-size:8pt; valign:top;}
    .Table {background:black}
    .TableHead { font:bold; color:white; background:black}
    .TableColumnHead { font:normal 'Verdana' bold; color:white;background:#6699CC}
    .TableRow {font:x-small 'Verdana'; color:black;background:#99CCFF}
  </STYLE>
<%@ Language=VBScript %>
<!-- #Include File="VeritabaninaBaglan.ASP" -->
  </HEAD>
<BODY bgcolor="#99CCFF">
<%
  Dim ad, soyad
  Client= Request.ServerVariables("HTTP_USER_AGENT")
  ClientVer=Split(Client, ";")
  ad=Request.QueryString("ad")
  soyad=Request.QueryString("soyad")

  rs.Open ("select * from t_calisan where ad='" & ad & "' and soyad='" & soyad & """"), con, 1
  if (rs.eof) then
    con.Execute "insert into t_Calisan (ad, soyad) values ( '" & ad & "' , '" & soyad & ""'"
    response.write ad & " " & soyad & " bilgisi eklendi...."
  else
    response.write ad & " " & soyad & " bilgisi veritaban&#305;na önceden
eklenmi&#351;...."
  end if

  rs.close
  con.close

  set rs=Nothing
  set con=Nothing %>

<center>
<br>
<table border="0">
<tr>
<td colspan=2 align=center>
<button onclick="window.close()">Kapat</button>
</td></tr></table>
</center></body>
</html>

```



*EkleKurum.asp*

```

<HTML>
  <HEAD> <TITLE>HKK - Kurum Ekleme</TITLE> </HEAD>
  <STYLE>
    body {font-family:verdana,helvetica;font-size:8pt}
    button {background-color:"#001964";font-
family:verdana,helvetica;color="WHITE";font-size:8pt;width:70}
    input {font-family:verdana,helvetica;font-size:8pt}
    Select {font-family:verdana,helvetica;font-size:8pt; valign:top;}
    .Table {background:black}
    .TableHead { font:bold; color:white; background:black}
    .TableColumnHead { font:normal 'Verdana' bold; color:white;background:#6699CC}
    .TableRow {font:x-small 'Verdana'; color:black;background:#99CCFF}
  </STYLE>

<%@ Language=VBScript %>
<!-- #Include File="VeritabaninaBaglan.ASP" -->

<FORM METHOD="post" id=frmEkle name=frmEkle>
<BODY bgcolor="#ffff00">

<center>
<table border="0">
<tr>
  <td width=30>Kurum:</td>
  <td><input type="text" name="txtKurum" size="30" ></td>
</tr>
<tr>
<td colspan=2 align=center>
<button onclick="Ekle()">Kaydet</button>

<button onclick="window.close()">Kapat</button>
</td>
</tr>
</table>

</form>
</center>

</BODY>

<script language="Javascript">
var mywin
var prevdiv="dummydiv"


function Ekle()
{
var str, i;

```

```
i=frmEkle.elements('txtKurum').value ;
if (i!=null)
{ str="ekleKurum2.asp?kurum=" + i; }

if (mywin!=null)
{
  if (!mywin.closed)
  {
    mywin.close();
    mywin=null;
  }
}
mywin=window.open(str,null,"directories=no,location=no,menubar=no,status,toolbar=no,r
esizable,scrollbars");
window.event.cancelBubble=true;
}

</script>
</HTML>
```



*EkleKurum2.asp*

```

<HTML>
  <HEAD> <TITLE>HKK - Kurum Ekleme </TITLE>
  <STYLE>
    body {font-family:verdana,helvetica;font-size:8pt}
    button {background-color:"#001964";font-
family:verdana,helvetica;color="WHITE";font-size:8pt;width:70}
    input {font-family:verdana,helvetica;font-size:8pt}
    Select {font-family:verdana,helvetica;font-size:8pt; valign:top;}
    .Table {background:black}
    .TableHead { font:bold; color:white; background:black}
    .TableColumnHead { font:normal 'Verdana' bold; color:white;background:#6699CC}
    .TableRow {font:x-small 'Verdana'; color:black;background:#99CCFF}
  </STYLE>
  <%@ Language=VBScript %>
  <!-- #Include File="VeritabaninaBaglan.ASP" -->
  </HEAD>
  <BODY bgcolor="#ffff00">
  <%

    Dim ad
    Client= Request.ServerVariables("HTTP_USER_AGENT")
    ClientVer=Split(Client, ";")
    ad=trim(Request.QueryString("kurum"))

    rs.Open ("select * from t_kurum where ad='" & ad & "'"), con, 1
    if (rs.eof) then
      con.Execute "insert into t_kurum (ad) values ('" & ad & "' ) "
      response.write ad & " " & " bilgisi eklendi...."
    else
      response.write ad & " bilgisi veritabanına önceden eklenmiş;...."
    end if

    rs.close
    con.close
    set rs=Nothing
    set con=Nothing %>

  <center>
  <br>
  <table border="0">
  <tr>
  <td colspan=2 align=center>
  <button onclick="window.close()">Kapat</button>
  </td>
  </tr>
  </table>
  </center>
  </body></html>

```

*EkleMal.asp*

```

<HTML>
  <HEAD> <TITLE>HKK - Mal Ekleme </TITLE>
  <STYLE>
    body {font-family:verdana,helvetica;font-size:8pt}
    button {background-color:"#001964";font-
family:verdana,helvetica;color="WHITE";font-size:8pt;width:70}
    input {font-family:verdana,helvetica;font-size:8pt}
    Select {font-family:verdana,helvetica;font-size:8pt; valign:top;}
    .Table {background:black}
    .TableHead { font:bold; color:white; background:black}
    .TableColumnHead { font:normal 'Verdana' bold; color:white;background:#6699CC}
    .TableRow {font:x-small 'Verdana'; color:black;background:#99CCFF}
  </STYLE>

  <%@ Language=VBScript %>
  <!-- #Include File="VeritabaninaBaglan.ASP" -->

  <FORM METHOD="post" id=frmEkle name=frmEkle>
  </HEAD>
  <BODY bgcolor="#904eff">

  <center>
  <table border="0">
  <tr>
    <td width=60>UniversalKod :</td>
    <td><input type="text" name="txtUniversalKod" size="5" ></td>
  </tr>
  <tr>
    <td width=60>Tan&#305;m :</td>
    <td><input type="text" name="txtTanim" size="30" </td>
  </tr>
  <tr>
    <td width=60>Birim :</td>
    <td><input type="text" name="txtBr" size="2" value="EA" </td>
  </tr>

  <tr>
  <td colspan=2 align=center>
  <button onclick="Ekle()">Kaydet</button>

  <button onclick="window.close()">Kapat</button>
  </td>
  </tr>
  </table>

  </form>
  </center>

```

```
</BODY>
```

```
<script language="Javascript">
```

```
var mywin
```

```
var prevdiv="dummydiv"
```

```
function Ekle()
```

```
{
```

```
var str, i, j, k;
```

```
    i=frmEkle.elements('txtUniversalKod').value ;
```

```
    j=frmEkle.elements('txtTanim').value ;
```

```
    k=frmEkle.elements('txtBr').value ;
```

```
    if (i!=null && j!=null)
```

```
        {str="ekleMal2.asp?kod=" + i + "&tanim=" + j + "&br=" + k;}
```

```
    if (mywin!=null)
```

```
    {
```

```
        if (!mywin.closed)
```

```
        {
```

```
            mywin.close();
```

```
            mywin=null;
```

```
        }
```

```
    }
```

```
    mywin=window.open(str,null,"directories=no,location=no,menubar=no,status,toolbar=no,resizeable,scrollbars");
```

```
    window.event.cancelBubble=true;
```

```
}
```

```
</script>
```

```
</HTML>
```

**EkleMal2.asp**

```

<HTML>
  <HEAD> <TITLE>HKK - Mal Ekleme </TITLE>
  <STYLE>
    body {font-family:verdana,helvetica;font-size:8pt}
    button {background-color:"#001964";font-
family:verdana,helvetica;color="WHITE";font-size:8pt;width:70}
    input {font-family:verdana,helvetica;font-size:8pt}
    Select {font-family:verdana,helvetica;font-size:8pt; valign:top;}
    .Table {background:black}
    .TableHead { font:bold; color:white; background:black}
    .TableColumnHead { font:normal 'Verdana' bold; color:white;background:#6699CC}
    .TableRow {font:x-small 'Verdana'; color:black;background:#99CCFF}
  </STYLE>

  <%@ Language=VBScript %>
  <!-- #Include File="VeritabaninaBaglan.ASP" -->
  </HEAD>
  <BODY bgcolor="#904eff">
  <%

    Dim kod, tanim, br
    Client= Request.ServerVariables("HTTP_USER_AGENT")
    ClientVer=Split(Client,",")
    kod=trim(Request.QueryString("kod"))
    tanim=trim(Request.QueryString("tanim"))
    br=trim(Request.QueryString("br"))

    rs.Open ("select * from t_mal where universalKod='" & kod & """), con, 1
    if (rs.eof) then
      con.Execute "insert into t_mal (universalkod, tanim, br) values ('" & kod & "', '" & tanim
& "', '" & br & "')"
      response.write kod & " " & tanim & " bilgisi eklendi...."
    else
      response.write kod & " " & tanim & " bilgisi veritabanına önceden
eklenmi;...."
    end if

    rs.close
    con.close
    set rs=Nothing
    set con=Nothing
  %>
  <center>
  <br><table border="0">
  <tr><td colspan=2 align=center>
  <button onclick="window.close()">Kapat</button>
  </td></tr></table></center>
  </body>
  </html>

```

***HKKAlimEmri.asp***

```

<%@ Language=VBScript %>
<HTML>
  <HEAD> <TITLE>HKK - Al&#305;m Emri </TITLE>
  <STYLE>
    body {font-family:verdana,helvetica;font-size:8pt}
    button {background-color:"#001964";font-
family:verdana,helvetica;color="WHITE";font-size:8pt;width:70}
    input {font-family:verdana,helvetica;font-size:8pt}
    Select {font-family:verdana,helvetica;font-size:8pt; valign:top;}
    .Table {background:black}
    .TableHead { font:bold; color:white; background:black}
    .TableColumnHead { font:normal 'Verdana' bold; color:white;background:#6699CC}
    .TableRow {font:x-small 'Verdana'; color:black;background:#99CCFF}
  </STYLE>

</HEAD>

<BODY LANGUAGE=javascript onerror="return window_onerror()">

<!-- #Include File="VeritabaninaBaglan.ASP" -->
<FORM METHOD=post ACTION="HKKAlimEmri2.asp" id=frmAlEmIst
name=frmAlEmIst>

<table border="0" cellspacing="0" cellpadding="20" height="200" width="95%"
class="Table">
<tr class="TableRow">
  <td width="10%" bgcolor="WHITE">
    &nbsp;
  </td>
  <td bgcolor="LIGHTYELLOW">
    <a href="default.asp"> HKK </a>
  </td>
  <td bgcolor="#99CCFF">
    <a href="calisan.asp">Çal&#305;&#351;an</a>
  </td>
  <td bgcolor="#904eff">
    <a href="mal.asp"> Mal </a>
  </td>
  <td bgcolor="#ffff00">
    <a href="kurum.asp"> Kurum </a>
  </td>
  <td bgcolor="#669900">
    <a href="hkkalimEmri.asp"> Al&#305;m Emri </a>
  </td>
  <td bgcolor="#CC0066">
    <a href="teklif.asp"> Teklifler </a>
  </td>
</tr>

```

```

</td>

<td width="10%" bgcolor="WHITE">
  &nbsp;
</td>
</tr>
<tr class="TableRow">
<td colspan="8" height="175" bgcolor="#669900" valign="TOP" >
<H1 align=center>Al&#305;m Emri</H1>
<font face="Arial" SIZE="2">

<%
Dim tarih, gun, ay, yil

gun = day(now())
ay = month(now())
yil=year(now())
if ( len(gun) = 1 ) then
  gun="0" & gun
end if
if ( len(ay) = 1 ) then
  ay="0" & ay
end if
tarih = yil & "-" & ay & "-" & gun
%>

<TABLE align=right>
  <TR>
    <TD>Tarih:<INPUT NAME="txtTarih" Id="txtTarih" maxLength="10" size="10" Value =
'<%=tarih %>' > </TD>
  </TR>
</TABLE>
<BR><br>

<%
Dim i, tumMal

Client= Request.ServerVariables("HTTP_USER_AGENT")
ClientVer=Split(Client,",")
rs.Open ("select * from t_calisan order by ad, soyad"), con, 1%>

<TABLE align=center>
  <tr >
    <TD VAlign=center>Al&#305;m Emrini Verecek Çal&#305;&#351;an: &nbsp; <select
size='1' name='selCalisan' onchange=getNumara() >
    <% Do While Not rs.EOF%>

      <option value='<%=Trim(rs("CalisanNo")) %>'> <%= Trim(rs("Ad")) %> &nbsp;&nbsp;<%=
= Trim(rs("Soyad")) %></option>
    <% rs.MoveNext
  Loop %>

```



```

        </select></td></tr>
    <tr>
        <td><input type='hidden' NAME="txtCalisanNo" Id="txtCalisanNo" maxLength = "5"
size = "5"></td>
    </tr>
</table>

<br>
<table align=center>
    <tr>
        <td>Mal</td>
        <td>Adet</td>
    </tr>

<%
rs.close
rs.Open ("select * from t_mal order by tanim"), con, 1

tumMal=""
Do While Not rs.EOF
    TumMal = TumMal & "<option Value=" & chr(34) & Trim(rs("MalNo")) & chr(34)
&">"
    TumMal = TumMal & Trim(rs("Tanim")) & "</option>"
    rs.MoveNext
Loop
rs.close
con.close

For i = 1 to 10
    Response.Write "<tr><td>"
    Response.Write "<select Language=" & chr(34) & "JavaScript" & chr(34) & " "
    Response.Write "Name=" & chr(34) & "Mal" & i & chr(34) & " "
    Response.Write "Id=" & chr(34) & "Mal" & i & chr(34) & ">"
    Response.Write "<option value=" & chr(34) & chr(34) & " selected='1'></option>" &
TumMal
    Response.Write "</select></td>"

    Response.Write "<td><input NAME=" & chr(34) & "txtAdet" & i & chr(34) & " "
    Response.Write "Id=" & chr(34) & "txtAdet" & i & chr(34)
    Response.Write " size=5 maxlength=5></td>"
    Response.Write "</tr>"
Next

'Memory icin
set con = Nothing
set rs = Nothing
%>
</table>

<p align="center">


```

```
<input type="submit" name="cmdSubmit" id="cmdSubmit" value="Al&#305;m Emrini
Ver">
</P>

</td>
</tr>
</table>

</FORM>
</BODY>
<%
Response.Write "<SCRIPT ID=getNumara LANGUAGE=Javascript>" & vbCRLF

Response.Write "function getNumara() {" & vbCRLF
Response.Write " " & vbCRLF
Response.Write "frmAlEmIst.elements('txtCalisanNo').value=
frmAlEmIst.elements('selCalisan').value" & vbCRLF
Response.Write "}" & vbCRLF & "</SCRIPT>"
%>
</HTML>
```



***HKKAlimEmri2.asp***

```

<%@ Language=VBScript %>
<HTML>
  <HEAD> <TITLE>HKK - Al&#305;m Emri </TITLE>
  <STYLE>
    body {font-family:verdana,helvetica;font-size:8pt}
    button {background-color:"#001964";font-
family:verdana,helvetica;color="WHITE";font-size:8pt;width:70}
    input {font-family:verdana,helvetica;font-size:8pt}
    Select {font-family:verdana,helvetica;font-size:8pt; valign:top;}
    .Table {background:black}
    .TableHead { font:bold; color:white; background:black}
    .TableColumnHead { font:normal 'Verdana' bold; color:white;background:#6699CC}
    .TableRow {font:x-small 'Verdana'; color:black;background:#99CCFF}
  </STYLE>

</HEAD>

<BODY LANGUAGE=javascript onerror="return window_onerror()">

<!-- #Include File="VeritabaninaBaglan.ASP" -->
<FORM METHOD=post id=frmCalisan name=frmCalisan>

<table border="0" cellspacing="0" cellpadding="20" height="200" width="95%"
class="Table">
<tr class="TableRow">
  <td width="10%" bgcolor="WHITE">
    &nbsp;
  </td>
  <td bgcolor="LIGHTYELLOW">
<a href="default.asp"> HKK </a>
  </td>
  <td bgcolor="#99CCFF">
<a href="calisan.asp">Çal&#305;&#351;an </a>
  </td>
  <td bgcolor="#904eff">
<a href="mal.asp"> Mal </a>
  </td>
  <td bgcolor="#ffff00">
<a href="kurum.asp"> Kurum </a>
  </td>
  <td bgcolor="#669900">
<a href="hkkalimEmri.asp"> Al&#305;m Emri </a>
  </td>
  <td bgcolor="#CC0066">
<a href="teklif.asp"> Teklifler </a>
  </td>

```

```

    <td width="10%" bgcolor="WHITE">
    &nbsp;
    </td>
</tr>
<tr class="TableRow">
<td colspan="8" height="175" bgcolor="#669900" valign="TOP" >
<H1 align=center>Al&#305;m Emri</H1>
<font face="Arial" SIZE="2">

```

```
<%
```

```

Dim bselCalisan, bTarih, bMal1
Dim XMLCode, sqlCode, AlEmNo
Dim fsObject
Dim i, j, kutuk

```

```

Client= Request.ServerVariables("HTTP_USER_AGENT")
ClientVer=Split(Client, ";")

```

```

bselCalisan = false
bTarih = false
bMal1 = false

```

```

If Request("txtTarih") > "" Then bTarih = True
If Request("txtCalisanNo") > "" Then bselCalisan = True
If Request("Mal1") > "" Then bMal1 = True
if (not bTarih) Then
    Response.write "Tarih girilmedi..."
end if
if (not bselCalisan) Then
    Response.write "Çal&#305;&#351;an seçilmedi..."
end if
if (not bMal1) Then
    Response.write "Mal seçilmedi..."
end if
if(Request("txtAdet1") = "") then
    Response.write "Adet bilgisi girilmedi..."
end if

```

```

If (bTarih AND bselCalisan AND bMal1 AND (Request("txtAdet1") > "")) Then
    response.write Request("txtCalisanNo") & " numaral&#305; çal&#305;&#351;an için "
    rs.Open ("select * from t_alEmNumara"), con, 1
    AlEmNo=rs("AlEmNo")+1
    con.Execute "update t_alEmNumara set alEmNo=" & AlEmNo & ""
    rs.close

```

```

XMLCode="<AlEmIst>"
XMLCode=XMLCode & "<Baslik Tarih=" & chr(34) & Request("txtTarih") & chr(34)
XMLCode=XMLCode & " AlEmNo=" & chr(34) & AlEmNo & chr(34)
XMLCode=XMLCode & " KurumNo=" & chr(34) & "1" & chr(34) & "/>"

rs.Open("select * from t_calisan where calisanNo=" & Request("txtCalisanNo")), con, 1
XMLCode=XMLCode & "<CalisanBilgisi Soyad=" & chr(34) & trim(rs("Soyad")) &
chr(34)
XMLCode=XMLCode & " Ad=" & chr(34) & trim(rs("ad")) & chr(34) & " CalisanNo="
XMLCode=XMLCode & chr(34) & trim(rs("calisanNo")) & chr(34) & "/>"
rs.close
j=0
For i = 1 to 10
    If Request("Mal" & i) > "" And Request("txtAdet" & i) > "" Then
        rs.Open("select * from t_mal where malNo=" & Request("Mal" & i) & ""), con, 1
        XMLCode=XMLCode & "<Mal Tanim=" & chr(34) & trim(rs("Tanim")) & chr(34)
        XMLCode=XMLCode & " UniversalKod=" & chr(34) & trim(rs("UniversalKod")) &
chr(34)
        XMLCode=XMLCode & " OlcuBirimi=" & chr(34) & trim(rs("Br")) & chr(34) & "
Adet="
        XMLCode=XMLCode & chr(34) & Request("txtAdet" & i) & chr(34) & " SatirNo="
        XMLCode=XMLCode & chr(34) & i & chr(34) & "/>"
        sqlCode="insert into t_AlEmriAyrinti(AlEmNo, UniversalKod, satirNo, Adet)"
        sqlCode=sqlCode & " values (" & alEmNo & ", " & trim(rs("UniversalKod")) & ", "
& i & ", "
        sqlCode=sqlCode & Request("txtAdet" & i) & ") "
        con.Execute sqlCode

        sqlCode="insert into t_GlnTklfAyrinti(AlEmNo, UniversalKod, satirNo, Adet)"
        sqlCode=sqlCode & " values (" & alEmNo & ", " & trim(rs("UniversalKod")) & ", "
& i & ", "
        sqlCode=sqlCode & Request("txtAdet" & i) & ") "
        con.Execute sqlCode
        rs.Close
        j=j+1 'mal sayisi
    End If
Next
rs.Open ("select * from t_alEmri where 1=2"), con, 1
sqlCode="insert into t_AlEmri (alEmNo, calisanNo, tarih, KacKalemMal)"
sqlCode=sqlCode & " values (" & alEmNo & ", " & Request("txtCalisanNo") & ", "
sqlCode=sqlCode & Request("txtTarih") & ", " & j & ") "
con.Execute sqlCode
rs.close
rs.Open ("select * from t_GelenTeklifler where 1=2"), con, 1
sqlCode="insert into t_GelenTeklifler (alEmNo) values (" & alEmNo & ") "
con.Execute sqlCode
rs.close
con.close

XMLCode=XMLCode & "<Toplam KacKalemMal=" & chr(34) & j & chr(34) & "/>" &
"</AlEmIst>"

```

Response.Write XMLCode

```
set FSObject=Server.CreateObject("Scripting.FileSystemObject")
kutuk="d:\inetpub\wwwroot\tez\Schedule\HKK.xml"
set XMLKutuk=FSObject.CreateTextFile(kutuk,True)
if Err.Number <> 0 then
    Response.Write "XML dosyas&#305; olu&#351;turulamad&#305;."
else
    XMLKutuk.Write XMLCode
    if Err.Number <> 0 then
        Response.Write "Dosyaya yaz&#305;lamad&#305;."
    else
        XMLKutuk.Close
        set bsl = Server.CreateObject("basla.tezim")
        bsl.baslat()
        Response.write kutuk & " XML dosyas&#305; yarat&#305;ld&#305;."
    end if
end if
```

```
'memory icin
set con = Nothing
set rs = Nothing
set bsl = Nothing
set FSObject = Nothing
```

End If

%>

</td>

</tr>

</table>

</body>

</html>

**Kurum.asp**

```

<%@ Language=VBScript %>
<HTML>
  <HEAD> <TITLE>HKK - Kurum Bilgileri </TITLE>
  <STYLE>
    body {font-family:verdana,helvetica;font-size:8pt}
    button {background-color:"#001964";font-
family:verdana,helvetica;color="WHITE";font-size:8pt;width:70}
    input {font-family:verdana,helvetica;font-size:8pt}
    Select {font-family:verdana,helvetica;font-size:8pt; valign:top;}
    .Table {background:black}
    .TableHead { font:bold; color:white; background:black}
    .TableColumnHead { font:normal 'Verdana' bold; color:white;background:#6699CC}
    .TableRow {font:x-small 'Verdana'; color:black;background:#99CCFF}
  </STYLE>

</HEAD>

<BODY LANGUAGE=javascript onerror="return window_onerror()">

<!-- #Include File="VeritabaninaBaglan.ASP" -->
<FORM METHOD=post id=frmKurum name=frmKurum>

<table border="0" cellspacing="0" cellpadding="20" height="200" width="95%"
class="Table">
<tr class="TableRow">
  <td width="10%" bgcolor="WHITE">
    &nbsp;
  </td>
  <td bgcolor="LIGHTYELLOW">
    <a href="default.asp"> HKK </a>
  </td>
  <td bgcolor="#99CCFF">
    <a href="calisan.asp">Çalışanlar </a>
  </td>
  <td bgcolor="#904eff">
    <a href="mal.asp"> Mal </a>
  </td>
  <td bgcolor="#ffff00">
    <a href="kurum.asp"> Kurum </a>
  </td>
  <td bgcolor="#669900">
    <a href="hkkalimEmri.asp">Alınan Emri </a>
  </td>
  <td bgcolor="#CC0066">
    <a href="teklif.asp"> Teklifler </a>
  </td>

```

```

    <td width="10%" bgcolor="WHITE">
    &nbsp;
    </td>
</tr>
<tr class="TableRow">
<td colspan="8" height="175" bgcolor="#ffff00" valign="TOP" >
<H1 align=center>Kurum Bilgileri</H1>
<font face="Arial" SIZE="2">

<%
Dim i, j

Client= Request.ServerVariables("HTTP_USER_AGENT")
ClientVer=Split(Client,",")
rs.Open ("select * from t_kurum order by ad"), con, 1%>

<TABLE align=center class="Table">
<tr >
<TD VAlign=center> <select size='15' name='selKurum' onchange=getNumara() >
<% Do While Not rs.EOF%>

    <option value='<%=Trim(rs("KurumNo")) %>'> <%= Trim(rs("Ad")) %> </option>
    <% rs.MoveNext
Loop
rs.close
con.close
set rs=Nothing
set con=Nothing %>

</select></td></tr>
<tr>
<TD><INPUT type='hidden' NAME="txtKurumNo" Id="txtKurumNo" maxLength = "5"
size = "5"></TD>
</TR>
</table>

<table align=center border="0" bgcolor="#B0C4DE">
<tr>
<td><button id="addnew" onclick="Ekle()">Ekle</button></td>
<td><button id="duzelt" onclick="Duzelt()">Düzelt</button></td>
<td><button id="sil" onclick="Sil()">Sil</button></td>
</tr>
</table>

</td>
</tr>
</table>

```



```
<script language="Javascript">
```

```
var mywin
var prevdiv="dummydiv"
```

```
function getNumara()
{
var i;
i = frmKurum.elements('selKurum').value;
frmKurum.elements('txtKurumNo').value= i;
}
```

```
function Sil()
{var i, str;
i=frmKurum.elements('txtKurumNo').value ;
str="silKurum.asp?ComponentID="+i;


if (mywin!=null)
{
if (!mywin.closed)
{
mywin.close();
mywin=null;
}
}
if (i != "")
{
mywin=window.open(str,null,"directories=no,location=no,menubar=no,status,toolbar=no,
resizable,scrollbars");
window.event.cancelBubble=true;
}
}
```

```
function Ekle()
{
var str;
str="EkleKurum.asp";
if (mywin!=null)
{
if (!mywin.closed)
{
mywin.close();
mywin=null;
}
}
mywin=window.open(str,null,"directories=no,location=no,menubar=no,status,toolbar=no,
resizable,scrollbars");
window.event.cancelBubble=true;
}
function Duzelt()
{
```

```
var str, i;
i=frmKurum.elements('txtKurumNo').value ;
str="duzeltKurum.asp?ComponentID="+i;

if (mywin!=null)
{
  if (!mywin.closed)
  {
    mywin.close();
    mywin=null;
  }
}
if (i != "")
{
  mywin=window.open(str,null,"directories=no,location=no,menubar=no,status,toolbar=no,resizable,scrollbars");
  window.event.cancelBubble=true;
}
}

</script>
</html>
```



**Mal.asp**

```

<%@ Language=VBScript %>
<HTML>
  <HEAD> <TITLE>HKK - Mal Bilgileri </TITLE>
  <STYLE>
    body {font-family:verdana,helvetica;font-size:8pt}
    button {background-color:"#001964";font-
family:verdana,helvetica;color="WHITE";font-size:8pt;width:70}
    input {font-family:verdana,helvetica;font-size:8pt}
    Select {font-family:verdana,helvetica;font-size:8pt; valign:top;}
    .Table {background:black}
    .TableHead { font:bold; color:white; background:black}
    .TableColumnHead { font:normal 'Verdana' bold; color:white;background:#6699CC}
    .TableRow {font:x-small 'Verdana'; color:black;background:#99CCFF}
  </STYLE>

</HEAD>

<BODY LANGUAGE=javascript onerror="return window_onerror()">

<!-- #Include File="VeritabaninaBaglan.ASP" -->
<FORM METHOD=post id=frmMal name=frmMal>

<table border="0" cellspacing="0" cellpadding="20" height="200" width="95%"
class="Table">
<tr class="TableRow">
  <td width="10%" bgcolor="WHITE">
    &nbsp;
  </td>
  <td bgcolor="LIGHTYELLOW">
    <a href="default.asp"> HKK </a>
  </td>
  <td bgcolor="#99CCFF">
    <a href="calisan.asp"> Çal&#305;&#351;an</a>
  </td>
  <td bgcolor="#904eff">
    <a href="mal.asp"> Mal </a>
  </td>
  <td bgcolor="#ffff00">
    <a href="kurum.asp"> Kurum </a>
  </td>
  <td bgcolor="#669900">
    <a href="hkkalimEmri.asp"> Al&#305;m Emri </a>
  </td>
  <td bgcolor="#CC0066">
    <a href="teklif.asp"> Teklifler </a>
  </td>

```

```

    <td width="10%" bgcolor="WHITE">
    &nbsp;
    </td>
</tr>
<tr class="TableRow">
<td colspan="8" height="175" bgcolor="#904eff" valign="TOP" >
<H1 align=center>Mal Bilgileri</H1>
<font face="Arial" SIZE="2">

<%
Dim i, j

Client= Request.ServerVariables("HTTP_USER_AGENT")
ClientVer=Split(Client, ";")
rs.Open ("select * from t_mal ORDER BY malNo"), con, 1%>

<TABLE align=center class="Table">
  <tr >
    <TD VAlign=center> <select size='15' name='selMal' onchange=getNumara() >
    <% Do While Not rs.EOF%>

      <option value='<%=Trim(rs("MalNo")) %>'> <%= Trim(rs("UniversalKod")) %>
&nbsp;<%= Trim(rs("Tanim")) %></option>
      <% rs.MoveNext
    Loop
    rs.close
    con.close
    set rs=Nothing
    set con=Nothing %>

    </select></td></tr>
  <tr>
    <TD><INPUT type='hidden' NAME="txtMalNo" Id="txtMalNo" maxLength = "5" size =
"5"></TD>
  </TR>
</table>

<table align=center border="0" bgcolor="#B0C4DE">
<tr>
<td><button id="addnew" onclick="Ekle()">Ekle</button></td>
<td><button id="duzelt" onclick="Duzelt()">Düzeltil</button></td>
<td><button id="sil" onclick="Sil()">Sil</button></td>
</tr>
</table>

</td>
</tr>
</table>

```

```
<script language="Javascript">
```

```
var mywin
var prevdiv="dummydiv"
```

```
function getNumara()
{
var i;
i = frmMal.elements('selMal').value;
frmMal.elements('txtMalNo').value= i;
}
```


```
function Sil()
{var i, str;
i=frmMal.elements('txtMalNo').value ;
str="silMal.asp?ComponentID="+i;
```

```
if (mywin!=null)
{
if (!mywin.closed)
{
mywin.close();
mywin=null;
}
}
if (i != "")
{
mywin=window.open(str,null,"directories=no,location=no,menubar=no,status,toolbar=no,resizeable,scrollbars");
window.event.cancelBubble=true;
}
}
```

```
function Ekle()
{
var str;
str="EkleMal.asp";
if (mywin!=null)
{
if (!mywin.closed)
{
mywin.close();
mywin=null;
}
}
mywin=window.open(str,null,"directories=no,location=no,menubar=no,status,toolbar=no,resizeable,scrollbars");
window.event.cancelBubble=true;
}
```

```
function Duzelt()
{
var str, i;
i=frmMal.elements('txtMalNo').value ;
str="duzeltMal.asp?ComponentID="+i;

if (mywin!=null)
{
if (!mywin.closed)
{
mywin.close();
mywin=null;
}
}
if (i != "")
{
mywin=window.open(str,null,"directories=no,location=no,menubar=no,status,toolbar=no,r
esizable,scrollbars");
window.event.cancelBubble=true;
}
}
</script>
</html>
```



*SilCalisan.asp*

```

<HTML>
  <HEAD> <TITLE>HKK - Çal&#305;&#351;an Silme </TITLE>
  <STYLE>
    body {font-family:verdana,helvetica;font-size:8pt}
    button {background-color:"#001964";font-
family:verdana,helvetica;color="WHITE";font-size:8pt;width:70}
    input {font-family:verdana,helvetica;font-size:8pt}
    Select {font-family:verdana,helvetica;font-size:8pt; valign:top;}
    .Table {background:black}
    .TableHead { font:bold; color:white; background:black}
    .TableColumnHead { font:normal 'Verdana' bold; color:white;background:#6699CC}
    .TableRow {font:x-small 'Verdana'; color:black;background:#99CCFF}
  </STYLE>

  <%@ Language=VBScript %>
  <!-- #Include File="VeritabaninaBaglan.ASP" -->

  </HEAD>

  <BODY bgcolor="#99CCFF">
  <%

  Dim calNo

  calNo=Request.QueryString("ComponentID")
  Client= Request.ServerVariables("HTTP_USER_AGENT")
  ClientVer=Split(Client, ";")
  rs.Open ("select * from t_calisan where 1=2" ), con, 1
  con.Execute "delete t_Calisan WHERE calisanNo = "" & calNo & """,adCmdText +
adExecuteNoRecords
  response.write calNo & " numaral&#305; kay&#305;t silindi...."
  rs.close
  con.close
  set rs=Nothing
  set con=Nothing %>

  <center>
  <table border="0">
  <tr>
  <td colspan=2 align=center>
  <button onclick="window.close()">Kapat</button>
  </td>
  </tr>
  </table>
  </center>
  </BODY>
  </HTML>

```

*SilKurum.asp*

```

<HTML>
  <HEAD> <TITLE>HKK - Kurum Silme </TITLE>
  <STYLE>
    body {font-family:verdana,helvetica;font-size:8pt}
    button {background-color:"#001964";font-
family:verdana,helvetica;color="WHITE";font-size:8pt;width:70}
    input {font-family:verdana,helvetica;font-size:8pt}
    Select {font-family:verdana,helvetica;font-size:8pt; valign:top;}
    .Table {background:black}
    .TableHead { font:bold; color:white; background:black}
    .TableColumnHead { font:normal 'Verdana' bold; color:white;background:#6699CC}
    .TableRow {font:x-small 'Verdana'; color:black;background:#99CCFF}
  </STYLE>

  <%@ Language=VBScript %>
  <!-- #Include File="VeritabaninaBaglan.ASP" -->
  </HEAD>
  <BODY bgcolor="#ffff00">

  <%

    Dim kurumNo

    kurumNo=Request.QueryString("ComponentID")
    Client= Request.ServerVariables("HTTP_USER_AGENT")
    ClientVer=Split(Client,",")
    rs.Open ("select * from t_kurum where 1=2 "), con, 1
    con.Execute "delete t_kurum WHERE KurumNo = '" & kurumNo & "'" ,adCmdText +
adExecuteNoRecords
    response.write kurumNo & " numaral&#305; kay&#305;t silindi...."
    rs.close
    con.close
    set rs=Nothing
    set con=Nothing %>

  <center>

  <table border="0">
  <tr>
  <td colspan=2 align=center>
  <button onclick="window.close()">Kapat</button>
  </td>
  </tr>
  </table>
  </center>
  </BODY>
  </HTML>

```



*SilMal.asp*

```

<HTML>
  <HEAD> <TITLE>HKK - Mal Silme</TITLE>
  <STYLE>
    body {font-family:verdana,helvetica;font-size:8pt}
    button {background-color:"#001964";font-
family:verdana,helvetica;color="WHITE";font-size:8pt;width:70}
    input {font-family:verdana,helvetica;font-size:8pt}
    Select {font-family:verdana,helvetica;font-size:8pt; valign:top;}
    .Table {background:black}
    .TableHead { font:bold; color:white; background:black}
    .TableColumnHead { font:normal 'Verdana' bold; color:white;background:#6699CC}
    .TableRow {font:x-small 'Verdana'; color:black;background:#99CCFF}
  </STYLE>

<%@ Language=VBScript %>
<!-- #Include File="VeritabaninaBaglan.ASP" -->

</HEAD>

<BODY bgcolor="#904eff">
<%

Dim malNo

malNo=Request.QueryString("ComponentID")
Client= Request.ServerVariables("HTTP_USER_AGENT")
ClientVer=Split(Client,",")
rs.Open ("select * from t_mal where l=2" ), con, 1
con.Execute "delete t_mal WHERE Malno = " & malNo & """,adCmdText +
adExecuteNoRecords
response.write malNo & " numaral&#305; kay&#305;t silindi...."
rs.close
con.close
set rs=Nothing
set con=Nothing %>

<center>
<table border="0">
<tr>
<td colspan=2 align=center>
<button onclick="window.close()">Kapat</button>
</td>
</tr>
</table>
</center>
</BODY>
</HTML>

```

**Teklif.asp**

```

<%@ Language=VBScript %>
<HTML>
  <HEAD> <TITLE>HKK - Teklif Bilgileri </TITLE>
  <STYLE>
    body {font-family:verdana,helvetica;font-size:8pt}
    button {background-color:"#001964";font-
family:verdana,helvetica;color="WHITE";font-size:8pt;width:70}
    input {font-family:verdana,helvetica;font-size:8pt}
    Select {font-family:verdana,helvetica;font-size:8pt; valign:top;}
    .Table {background:black}
    .TableHead { font:bold; color:white; background:black}
    .TableColumnHead { font:normal 'Verdana' bold; color:white;background:#6699CC}
    .TableRow {font:x-small 'Verdana'; color:black;background:#99CCFF}
  </STYLE>

</HEAD>

<BODY LANGUAGE=javascript onerror="return window_onerror()">

<!-- #Include File="VeritabaninaBaglan.ASP" -->

<FORM METHOD="post" id=frmTeklif name=frmTeklif>

<table border="0" cellspacing="0" cellpadding="20" height="200" width="95%"
class="Table">
<tr class="TableRow">
  <td width="10%" bgcolor="WHITE">
    &nbsp;
  </td>
  <td bgcolor="LIGHTYELLOW">
    <a href="default.asp"> HKK </a>
  </td>
  <td bgcolor="#99CCFF">
    <a href="calisan.asp">Çal&#305;&#351;an </a>
  </td>
  <td bgcolor="#904eff">
    <a href="mal.asp"> Mal </a>
  </td>
  <td bgcolor="#ffff00">
    <a href="kurum.asp"> Kurum </a>
  </td>
  <td bgcolor="#669900">
    <a href="hkkalimEmri.asp"> Al&#305;m Emri </a>
  </td>
  <td bgcolor="#CC0066">
    <a href="teklif.asp"> Teklifler </a>
  </td>
</tr>

```

```

</td>

<td width="10%" bgcolor="WHITE">
  &nbsp;
</td>
</tr>
<tr class="TableRow">
<td colspan="8" height="175" bgcolor="#CC0066" valign="TOP" >
<H1 align=center>Teklif Bilgileri</H1>
<font face="Arial" SIZE="2">

```

```
<%
```

```
Dim i, j, rs2, rs3, rs4, sqlCode, firma, calisan
```

```

Client= Request.ServerVariables("HTTP_USER_AGENT")
ClientVer=Split(Client, ";")
set rs2=server.CreateObject("adodb.recordset")
set rs3=server.CreateObject("adodb.recordset")
set rs4=server.CreateObject("adodb.recordset")
sqlCode="select * from t_GelenTeklifler order by AEmNo"
rs.Open (sqlCode), con, 1%>

```

```

<TABLE CLASS="Table" align="center" border = 1>
  <Tr class="TableHead" align="center">
    <td class="TableColumnHead">Al&#305;m Emri No</td>
    <td class="TableColumnHead">Çal&#305;e&#351;an </td>
    <td class="TableColumnHead">Tarih</td>
    <td class="TableColumnHead">Kaç Kalem Mal</td>
    <td class="TableColumnHead">S1 Toplam</td>
    <td class="TableColumnHead">S2 Toplam</td>
    <td class="TableColumnHead">S3 Toplam</td>
    <td class="TableColumnHead">Sipari&#351; Verilen Kurum</td>
  </tr>

```

```
<%
```

```

Do while (Not rs.eof) %>
  <tr class="TableRow">
    <TD VAlign=top>
      <p align="right"><% = rs("AEmNo")%></TD>
    <%
      firma=rs("kazananFirma")
      rs4.Open ("select * from t_AEmri where alEmNo= " & rs("alEmNo") & """), con, 1
      calisan=rs4("calisanNo")
      rs2.Open ("select ad, soyad from t_calisan where calisanNo= " & calisan & """), con,

```

```
1
```

```

    %>
    <TD VAlign=top><% = trim(rs2("ad")) & " " & trim(rs2("soyad"))%></TD>
    <TD VAlign=top>
      <p align="center"><% = rs4("tarih")%></TD>

```

```

<TD VAlign=top>
  <p align="center"><% = rs4("kacKalemMal")%></TD>
<TD VAlign=top>
  <p align="right"><% = rs("S1Toplam")%></TD>
<TD VAlign=top>
  <p align="right"><% = rs("S2Toplam")%></TD>
<TD VAlign=top>
  <p align="right"><% = rs("S3Toplam")%></TD>

```

```

<% if (rs("S1Toplam")<>"0") AND (rs("S2Toplam")<>"0") AND
(rs("S3Toplam")<>"0") then

```

```

  rs3.open ("select ad from t_kurum where kurumNo= '" & firma & "'"), con, 1 %>
  <TD VAlign=top>
    <p align="center"><% = trim(rs3("ad"))%></TD>
    <% rs3.close
else %>
  <TD VAlign=top><% = " "%></TD>
<% end if%>

```

```

</tr>
  <% rs2.close
  rs4.close
  rs.MoveNext
Loop
%>

```

```

</TABLE>
<% rs.close
con.close
set rs=Nothing
set rs2=Nothing
set rs3=Nothing
set rs4=Nothing
set con=Nothing %>

```

```

<br>
<br>
<table border="0" align="center">
<tr>
  <td width="240"> Ayr&#305;nt&#305;s&#305;n&#305; görmek istedi&#287;iniz
  AEmNo :</td>
  <td><input type="text" name="txtAEmNo" size="5" value=" "></td>
</tr>
</table>
<table border="0" align="center">
<tr>
  <td><button onclick="Ayrinti()">Ayr&#305;nt&#305;</button></td>
  <td><button onclick="window.close()">Kapat</button></td>

```

```

</tr>
</TABLE>

```

```
</td>  
</tr>  
</table>
```

```
<script language="Javascript">  
var mywin  
var prevdiv="dummydiv"
```

```
function Ayrinti()
```

```
{  
var str, i;
```

```
    i=frmTeklif.elements('txtAIEmNo').value ;
```

```
    str="teklif2.asp?AIEmNo="+ i;
```

```
    if (mywin!=null)
```

```
    {
```

```
        if (!mywin.closed)
```

```
        {
```

```
            mywin.close();
```

```
            mywin=null;
```

```
        }
```

```
    }
```

```
    mywin=window.open(str,null,"directories=no,location=no,menubar=no,status,toolbar=no,r  
esizable,scrollbars");
```

```
    window.event.cancelBubble=true;
```

```
}
```

```
</script>
```

```
</html>
```

**Teklif2.asp**

```

<%@ Language=VBScript %>
<HTML>
  <HEAD> <TITLE>HKK - Teklif Bilgileri </TITLE>
  <STYLE>
    body {font-family:verdana,helvetica;font-size:8pt}
    button {background-color:"#001964";font-
family:verdana,helvetica;color="WHITE";font-size:8pt;width:70}
    input {font-family:verdana,helvetica;font-size:8pt}
    Select {font-family:verdana,helvetica;font-size:8pt; valign:top;}
    .Table {background:black}
    .TableHead { font:bold; color:white; background:black}
    .TableColumnHead { font:normal 'Verdana' bold; color:white;background:#6699CC}
    .TableRow {font:x-small 'Verdana'; color:black;background:#99CCFF}
  </STYLE>

</HEAD>

<BODY bgcolor="#CC0066" LANGUAGE=javascript onerror="return window_onerror()">

<!-- #Include File="VeritabaninaBaglan.ASP" -->

<FORM METHOD="post" id=frmTeklif2 name=frmTeklif2>

<%
  Dim alEmNo, rs2, sqlCode

  alEmNo=Request.QueryString("AlEmNo")
  Client= Request.ServerVariables("HTTP_USER_AGENT")
  ClientVer=Split(Client,",")
  sqlCode="select universalKod, Adet, S1Fiyat, S1HesaplanmisFiyat, S2Fiyat,"
  sqlCode=sqlCode & "S2HesaplanmisFiyat, S3Fiyat, S3HesaplanmisFiyat from"
  sqlCode=sqlCode & " t_GlnTklfAyrinti where alEmNo = '" & alEmNo & "' order by
  satirNo"
  rs.Open (sqlcode), con, 1
  set rs2=server.CreateObject("adodb.recordset")

%>

<TABLE CLASS="Table" align="center" border = 1 width="780">
  <Tr class="TableHead" align="center">
    <td class="TableColumnHead" width="58">Mal Tan&#305;m&#305;</td>

    <td class="TableColumnHead" width="41">Adet</td>
    <td class="TableColumnHead" width="57">Sat&#305;c&#305;1 Fiyat&#305;</td>

```

```

        <td class="TableColumnHead" width="154">Sat&#305;c&#305;1
Hesaplanm&#305;&#351;
        Fiyat</td>
        <td class="TableColumnHead" width="59">Sat&#305;c&#305;2 Fiyat&#305;</td>
        <td class="TableColumnHead" width="152">Sat&#305;c&#305;2
Hesaplanm&#305;&#351;
        Fiyat</td>
        <td class="TableColumnHead" width="57">Sat&#305;c&#305;3 Fiyat&#305;</td>
        <td class="TableColumnHead" width="152">Sat&#305;c&#305;3
Hesaplanm&#305;&#351;
        Fiyat</td>
</tr>

<%
    Do while (Not rs.eof) %>

        <tr class="TableRow">
            <%rs2.Open ("select tanim from t_mal where universalKod = "" &
rs("universalKod") & """), con, 1 %>
            <TD VAlign=top width="58"><% = rs2("tanim")%></TD>
            <% rs2.close %>

            <% For Index=1 to (rs.fields.count-1) %>
                <TD VAlign=top width="41"><% = rs(Index)%></TD>
            <% Next %>
        </tr>
        <% rs.MoveNext
    Loop
%>

</TABLE>
<% rs.close
con.close
set rs=Nothing
set con=Nothing %>
<br><br>
<center><button onclick="window.close()">Kapat</button></center>
</FORM>&nbsp;

</BODY>
</HTML>

```

***VeritabaninaBaglan.asp***

```
<!-- #Include File="ADOVBS.INC" -->
<%
Dim rs ,con, ConnectionString

ConnectionString = "dsn=HKK;server=STUDY;database=HKK;uid=HKKUser;pwd="
set con=server.CreateObject ("adodb.connection")
set rs=server.CreateObject("adodb.recordset")
con.Open connectionstring
Rs.ActiveConnection = Con
Rs.CursorLocation = adUseClient
set Cmd= server.CreateObject("adodb.command")
Cmd.ActiveConnection =ConnectionString
%>
```





**Ek 5 Hava Kuvvetleri Komutanlığı için COM Kodları****Basla.vbp**

Option Explicit

```
Private g_MSMTxDisp As msmq.MSMQTransactionDispenser
Private g_MSMQQueue As msmq.MSMQQueue
Private g_MSMQInfo As msmq.MSMQQueueInfo
Private g_SkedFile As String
Private g_DataFile As String
```

```
Private Sub Class_Initialize()
    Set g_MSMQInfo = CreateObject("MSMQ.MSMQQueueInfo")
    Set g_MSMTxDisp = CreateObject("MSMQ.MSMQTransactionDispenser")
    'schedule ve Alım Emrinin olduğu kutukler:
    g_SkedFile = "sked:/// " & "d:\inetpub\wwwroot\tez\Schedule\HKK.skx"
    g_DataFile = "d:\inetpub\wwwroot\tez\Schedule\HKK.xml"
End Sub
```

```
Private Sub Class_Terminate()
    Set g_MSMQInfo = Nothing
    Set g_MSMTxDisp = Nothing
    Set g_MSMQQueue = Nothing
End Sub
```

```
Public Sub baslat()
    Dim objfs As New FileSystemObject
    Dim objFileSys, strTimeNow, FileName, FileStream
    Dim doc As MSXML2.DOMDocument
```

```
On Error GoTo hata
```

```
'Connect To MSMQ and Remove Any Existing Messages
PurgeMSMQ "DIRECT=OS:\private$\HAIEmIst"
'Send Selected message to MSMQ
ExecuteMSMQ "DIRECT=OS:\private$\HAIEmIst", objfs.OpenTextFile(g_DataFile,
ForReading).ReadAll
```

```
Set doc = New MSXML2.DOMDocument
doc.Load g_DataFile
If doc.parseError.errorCode <> 0 Then 'The data was not valid XML
    Err.Raise doc.parseError.errorCode, "Tez Basla component", _
    "AlEmIst islenemiyor : " + _
    vbCrLf + vbCrLf + doc.parseError.reason + vbCrLf + " satir no: " + _
    CStr(doc.parseError.Line) + vbCrLf + "yeri: " + _
    CStr(doc.parseError.linepos)
End If
```

```
ExecuteSchedule g_SkedFile
```

```

FileName = "d:\inetpub\wwwroot\tez\logHKK.txt"
strTimeNow = CStr(FormatDateTime(Now(), 1)) & " / " & CStr(FormatDateTime(Now(),
3))
Set objFileSys = CreateObject("Scripting.FileSystemObject")
Set fileStream = objFileSys.OpenTextFile(FileName, ForAppending)
fileStream.WriteLine ""
fileStream.WriteLine strTimeNow & " ==> HKK.xml HKKAEmIst kuyruguna basari ile
atildi."
fileStream.WriteLine strTimeNow & " ==> Schedule basarili olarak calistirildi."
fileStream.Close

Set objfs = Nothing
Set objFileSys = Nothing
Set fileStream = Nothing
Set doc = Nothing

```

Exit Sub

hata:

```

MsgBox Err.Description & vbCrLf & "Error: " & Err.Number & " (0x" &
Hex(Err.Number) & ")", vbCritical, "Error " & Err.Source
Err.Clear
End Sub

```

Public Sub ExecuteMSMQ(ByVal strQueuePath As String, DataToQueue As String)

```

Dim QueueMsg As New MSMQMessage
Dim MSMQTx
Dim strData As String
Dim fSend As Boolean
Dim txt As TextStream
Dim mybyte() As Byte

```

```

On Error GoTo Err_SendMSMQ
g_MSMQInfo.FormatName = strQueuePath
Set g_MSMQQueue = g_MSMQInfo.Open(MQ_SEND_ACCESS, MQ_DENY_NONE)
mybyte = StrConv(DataToQueue, vbFromUnicode)
QueueMsg.Body = DataToQueue

```

```

Set MSMQTx = g_MSMTxDisp.BeginTransaction
QueueMsg.Send g_MSMQQueue, MSMQTx
MSMQTx.Commit

```

```

Set QueueMsg = Nothing
Set MSMQTx = Nothing
Exit Sub

```

Err\_SendMSMQ:

```

Err.Raise Err.Number, "HAEmIst'e mesaji gonderirken", "MSMQ'ya mesaj
gonderilemiyor "" & strQueuePath & ""." & vbCrLf & vbCrLf & Err.Description
Exit Sub
End Sub

```

```

Public Sub PurgeMSMQ(ByVal strQueuePath As String)
    Dim l_MSMQMsg As MSMQMessage

    On Error GoTo Err_ConnectMSMQ
    g_MSMQInfo.FormatName = strQueuePath
    Set g_MSMQQueue = g_MSMQInfo.Open(MQ_RECEIVE_ACCESS,
MQ_DENY_NONE)

    On Error GoTo Err_PurgeMSMQ
    Do
        Set l_MSMQMsg = g_MSMQQueue.Receive(, , 1)
    Loop While Not l_MSMQMsg Is Nothing

Exit Sub

Err_ConnectMSMQ:
    Err.Raise Err.Number, "Connecting To HAEmIst MSMQ", "Could Not Open the MSMQ
Queue """" & strQueuePath & """"." & vbCrLf & vbCrLf & Err.Description
    Exit Sub
Err_PurgeMSMQ:
    Err.Raise Err.Number, "Cleaning HAEmIst MSMQ", "Could Not Remove Existing
Messages from MSMQ Queue """" & strQueuePath & """"." & vbCrLf & vbCrLf &
Err.Description
    Exit Sub
End Sub

Public Sub ExecuteSchedule(ByVal strSchedule)
    Dim SendPAQ As Object
    On Error GoTo Err_ExecSched

    Set SendPAQ = GetObject(strSchedule)
    If SendPAQ Is Nothing Then
        Err.Raise vbObjectError + 1, , "Invalid Schedule Handle Returned."
    End If
    Set SendPAQ = Nothing
    Exit Sub

Err_ExecSched:
    Err.Raise Err.Number, "Starting Schedule", "Could Not Launch the XLANG Schedule" &
vbCrLf & "Please verify the path to the SKX file and path to your data are correct. Also
make sure the private queues have been created." & vbCrLf & vbCrLf & Err.Description
    Exit Sub
End Sub

```

**HKKSprS1.vbp**

Implements IBTSAppIntegration

Option Explicit

Private rs, con, cmd, ConnectionString

Private m\_DOMDocument As MSXML2.DOMDocument

\*\*\*\*\*

\*\*\*\*\* Gelen Dokuman: T\_Fatura.xml, BizTalk Serverdan alınır.

\*\*\*\*\*

Private Function IBTSAppIntegration\_ProcessMessage(  
ByVal bstrDocument As String) As String

Dim j As Integer

Dim alEmNo As Integer

Dim objFileSys, strTimeNow, FileName, fileStream, sqlCode, alEmToplami

On Error GoTo hata

ConnectionString = "dsn=HKK;server=STUDY;database=HKK;uid=HKKUser;pwd="

Set con = CreateObject("adodb.connection")

Set rs = CreateObject("adodb.recordset")

con.Open ConnectionString

rs.ActiveConnection = con

rs.CursorLocation = adUseClient

Set cmd = CreateObject("adodb.command")

cmd.ActiveConnection = ConnectionString

Set m\_DOMDocument = New MSXML2.DOMDocument

m\_DOMDocument.async = False

IBTSAppIntegration\_ProcessMessage = ""

m\_DOMDocument.loadXML bstrDocument

If m\_DOMDocument.parseError.errorCode <> 0 Then 'The data was not valid XML

Err.Raise m\_DOMDocument.parseError.errorCode, "Satici1 Siparis component", \_

"Gelen dokuman islenemiyor :" + \_

vbCrLf + vbCrLf + m\_DOMDocument.parseError.reason + vbCrLf + " satir no: " + \_

CStr(m\_DOMDocument.parseError.Line) + vbCrLf + "yeri: " + \_

CStr(m\_DOMDocument.parseError.linepos)

End If

j = m\_DOMDocument.documentElement.childNodes.length

alEmNo =

m\_DOMDocument.documentElement.childNodes.Item(0).Attributes.Item(2).Text

m\_DOMDocument.documentElement.childNodes.Item(j -  
2).childNodes.Item(0).Attributes.Item(0).Text = "Hava Kuvvetleri K.ligi"

m\_DOMDocument.documentElement.childNodes.Item(j -  
2).childNodes.Item(0).Attributes.Item(1).Text = "Merkez"

```

m_DOMDocument.documentElement.childNodes.Item(j -
2).childNodes.Item(0).Attributes.Item(3).Text = "ANKARA"
m_DOMDocument.documentElement.childNodes.Item(j -
2).childNodes.Item(0).Attributes.Item(4).Text = "06000"
m_DOMDocument.documentElement.childNodes.Item(j -
1).childNodes.Item(0).Attributes.Item(0).Text = "Hava Harp Okulu K.ligi"
m_DOMDocument.documentElement.childNodes.Item(j -
1).childNodes.Item(0).Attributes.Item(1).Text = "Yesilyurt"
m_DOMDocument.documentElement.childNodes.Item(j -
1).childNodes.Item(0).Attributes.Item(3).Text = "ISTANBUL"
m_DOMDocument.documentElement.childNodes.Item(j -
1).childNodes.Item(0).Attributes.Item(4).Text = "34800"

```

```

alEmToplami = m_DOMDocument.documentElement.childNodes.Item(j -
3).Attributes.Item(1).Text

```

```

sqlCode = "select * from t_GelenTeklifler where 1=2"
con.BeginTrans
rs.Open sqlCode, con, 1

```

```

sqlCode = "update t_GelenTeklifler set alEmToplami="
sqlCode = sqlCode & alEmToplami & ", kazananFirma='3' where "
sqlCode = sqlCode & "alEmno=" & alEmNo & ""
con.Execute sqlCode
rs.Close
con.CommitTrans
con.Close

```

```

FileName = "d:\inetpub\wwwroot\tez\SiparisS1\Sprs*.xml"
strTimeNow = Left(Replace(CStr(FormatDateTime(Now(), 3)), ":", ""), 6)
Set objFileSys = CreateObject("Scripting.FileSystemObject")
Set fileStream = objFileSys.CreateTextFile(Replace(FileName, "*", strTimeNow), False)
fileStream.WriteLine m_DOMDocument.xml
fileStream.Close

```

```

FileName = "d:\inetpub\wwwroot\tez\Satici1\Siparis\Sprs*.xml"
strTimeNow = Left(Replace(CStr(FormatDateTime(Now(), 3)), ":", ""), 6)
Set objFileSys = CreateObject("Scripting.FileSystemObject")
Set fileStream = objFileSys.CreateTextFile(Replace(FileName, "*", strTimeNow), False)
fileStream.WriteLine m_DOMDocument.xml
fileStream.Close

```

```

FileName = "d:\inetpub\wwwroot\tez\logHKK.txt"
strTimeNow = CStr(FormatDateTime(Now(), 1)) & " / " & CStr(FormatDateTime(Now(),
3))
Set objFileSys = CreateObject("Scripting.FileSystemObject")
Set fileStream = objFileSys.OpenTextFile(FileName, ForAppending)
fileStream.WriteLine strTimeNow & " ==> HKKSprsS1 modülü ile " & alEmNo & "
numarali Alim Emri için Satici le siparis verildi."
fileStream.Close

```

```
IBTSAppIntegration_ProcessMessage = m_DOMDocument.xml  
Set con = Nothing  
Set rs = Nothing  
Set cmd = Nothing  
Set m_DOMDocument = Nothing  
Set objFileSys = Nothing  
Set fileStream = Nothing  
Exit Function
```

hata:

```
MsgBox Err.Description & vbCrLf & "Error: " & Err.Number & " (0x" &  
Hex(Err.Number) & ")", vbCritical, "Error " & Err.Source  
Err.Clear
```

```
End Function
```



**HKKSprsS2.vbp**

Implements IBTSAppIntegration

Option Explicit

Private rs, con, cmd, ConnectionString

Private m\_DOMDocument As MSXML2.DOMDocument

\*\*\*\*\*

\*\*\*\*\* Gelen Dokuman: T\_Fatura.xml, BizTalk Serverdan alınır.

\*\*\*\*\*

Private Function IBTSAppIntegration\_ProcessMessage( \_  
ByVal bstrDocument As String) As String

Dim j As Integer

Dim alEmNo As Integer

Dim objFileSys, strTimeNow, FileName, fileStream, sqlCode, alEmToplami

On Error GoTo hata

ConnectionString = "dsn=HKK;server=STUDY;database=HKK;uid=HKKUser;pwd="

Set con = CreateObject("adodb.connection")

Set rs = CreateObject("adodb.recordset")

con.Open ConnectionString

rs.ActiveConnection = con

rs.CursorLocation = adUseClient

Set cmd = CreateObject("adodb.command")

cmd.ActiveConnection = ConnectionString

Set m\_DOMDocument = New MSXML2.DOMDocument

m\_DOMDocument.async = False

IBTSAppIntegration\_ProcessMessage = ""

m\_DOMDocument.loadXML bstrDocument

If m\_DOMDocument.parseError.errorCode <> 0 Then 'The data was not valid XML

Err.Raise m\_DOMDocument.parseError.errorCode, "Satici2 Siparis component", \_

"Gelen dokuman islenemiyor ." + \_

vbCrLf + vbCrLf + m\_DOMDocument.parseError.reason + vbCrLf + " satir no: " + \_

CStr(m\_DOMDocument.parseError.Line) + vbCrLf + "yeri: " + \_

CStr(m\_DOMDocument.parseError.linepos)

End If

j = m\_DOMDocument.documentElement.childNodes.length

alEmNo =

m\_DOMDocument.documentElement.childNodes.Item(0).Attributes.Item(2).Text

m\_DOMDocument.documentElement.childNodes.Item(j -

2).childNodes.Item(0).Attributes.Item(0).Text = "Hava Kuvvetleri K.ligi"

m\_DOMDocument.documentElement.childNodes.Item(j -

2).childNodes.Item(0).Attributes.Item(1).Text = "Merkez"

```

m_DOMDocument.documentElement.childNodes.Item(j -
2).childNodes.Item(0).Attributes.Item(3).Text = "ANKARA"
m_DOMDocument.documentElement.childNodes.Item(j -
2).childNodes.Item(0).Attributes.Item(4).Text = "06000"
m_DOMDocument.documentElement.childNodes.Item(j -
1).childNodes.Item(0).Attributes.Item(0).Text = "Hava Harp Okulu K.ligi"
m_DOMDocument.documentElement.childNodes.Item(j -
1).childNodes.Item(0).Attributes.Item(1).Text = "Yesilyurt"
m_DOMDocument.documentElement.childNodes.Item(j -
1).childNodes.Item(0).Attributes.Item(3).Text = "ISTANBUL"
m_DOMDocument.documentElement.childNodes.Item(j -
1).childNodes.Item(0).Attributes.Item(4).Text = "34800"

```

```

alEmToplami = m_DOMDocument.documentElement.childNodes.Item(j -
3).Attributes.Item(1).Text

```

```

sqlCode = "select * from t_GelenTeklifler where l=2"
con.BeginTrans
rs.Open sqlCode, con, 1

```

```

sqlCode = "update t_GelenTeklifler set alEmToplami="
sqlCode = sqlCode & alEmToplami & ", kazananFirma='4' where "
sqlCode = sqlCode & "alEmno=" & alEmNo & ""
con.Execute sqlCode
rs.Close
con.CommitTrans
con.Close

```

```

FileName = "d:\inetpub\wwwroot\tez\SiparisS2\Sprs*.xml"
strTimeNow = Left(Replace(CStr(FormatDateTime(Now(), 3)), ":", ""), 6)
Set objFileSys = CreateObject("Scripting.FileSystemObject")
Set fileStream = objFileSys.CreateTextFile(Replace(FileName, "*", strTimeNow), False)
fileStream.WriteLine m_DOMDocument.xml
fileStream.Close

```

```

FileName = "d:\inetpub\wwwroot\tez\Satici2\Siparis\Sprs*.xml"
strTimeNow = Left(Replace(CStr(FormatDateTime(Now(), 3)), ":", ""), 6)
Set objFileSys = CreateObject("Scripting.FileSystemObject")
Set fileStream = objFileSys.CreateTextFile(Replace(FileName, "*", strTimeNow), False)
fileStream.WriteLine m_DOMDocument.xml
fileStream.Close

```

```

FileName = "d:\inetpub\wwwroot\tez\logHKK.txt"
strTimeNow = CStr(FormatDateTime(Now(), 1)) & " / " & CStr(FormatDateTime(Now(),
3))
Set objFileSys = CreateObject("Scripting.FileSystemObject")
Set fileStream = objFileSys.OpenTextFile(FileName, ForAppending)
fileStream.WriteLine strTimeNow & " ==> HKKSprsS2 modülü ile " & alEmNo & "
numarali Alim Emri için Satici2ye siparis verildi."
fileStream.Close

```



```
IBTSAppIntegration_ProcessMessage = m_DOMDocument.xml  
Set con = Nothing  
Set rs = Nothing  
Set cmd = Nothing  
Set m_DOMDocument = Nothing  
Set objFileSys = Nothing  
Set fileStream = Nothing  
Exit Function
```

hata:

```
MsgBox Err.Description & vbCrLf & "Error: " & Err.Number & " (0x" &  
Hex(Err.Number) & ")", vbCritical, "Error " & Err.Source  
Err.Clear
```

End Function



**HKKSprsS3.vbp**

Implements IBTSAppIntegration

Option Explicit

Private rs, con, cmd, ConnectionString

Private m\_DOMDocument As MSXML2.DOMDocument

\*\*\*\*\*

\*\*\*\*\* Gelen Dokuman: T\_Fatura.xml, BizTalk Serverdan alınır.

\*\*\*\*\*

Private Function IBTSAppIntegration\_ProcessMessage( \_  
ByVal bstrDocument As String) As String

Dim j As Integer

Dim alEmNo As Integer

Dim objFileSys, strTimeNow, FileName, FileStream, sqlCode, alEmToplami

On Error GoTo hata

ConnectionString = "dsn=HKK;server=STUDY;database=HKK;uid=HKKUser;pwd="

Set con = CreateObject("adodb.connection")

Set rs = CreateObject("adodb.recordset")

con.Open ConnectionString

rs.ActiveConnection = con

rs.CursorLocation = adUseClient

Set cmd = CreateObject("adodb.command")

cmd.ActiveConnection = ConnectionString

Set m\_DOMDocument = New MSXML2.DOMDocument

m\_DOMDocument.async = False

IBTSAppIntegration\_ProcessMessage = ""

m\_DOMDocument.loadXML bstrDocument

If m\_DOMDocument.parseError.errorCode <> 0 Then 'The data was not valid XML

Err.Raise m\_DOMDocument.parseError.errorCode, "Satici3 Siparis component", \_

"Gelen dokuman islenemiyor : " + \_

vbCrLf + vbCrLf + m\_DOMDocument.parseError.reason + vbCrLf + " satir no: " + \_

CStr(m\_DOMDocument.parseError.Line) + vbCrLf + "yeri: " + \_

CStr(m\_DOMDocument.parseError.linepos)

End If

j = m\_DOMDocument.documentElement.childNodes.length

alEmNo =

m\_DOMDocument.documentElement.childNodes.Item(0).Attributes.Item(2).Text

m\_DOMDocument.documentElement.childNodes.Item(j -

2).childNodes.Item(0).Attributes.Item(0).Text = "Hava Kuvvetleri K.ligi"

m\_DOMDocument.documentElement.childNodes.Item(j -

2).childNodes.Item(0).Attributes.Item(1).Text = "Merkez"

```

m_DOMDocument.documentElement.childNodes.Item(j -
2).childNodes.Item(0).Attributes.Item(3).Text = "ANKARA"
m_DOMDocument.documentElement.childNodes.Item(j -
2).childNodes.Item(0).Attributes.Item(4).Text = "06000"
m_DOMDocument.documentElement.childNodes.Item(j -
1).childNodes.Item(0).Attributes.Item(0).Text = "Hava Harp Okulu K.ligi"
m_DOMDocument.documentElement.childNodes.Item(j -
1).childNodes.Item(0).Attributes.Item(1).Text = "Yesilyurt"
m_DOMDocument.documentElement.childNodes.Item(j -
1).childNodes.Item(0).Attributes.Item(3).Text = "ISTANBUL"
m_DOMDocument.documentElement.childNodes.Item(j -
1).childNodes.Item(0).Attributes.Item(4).Text = "34800"

```

```

alEmToplami = m_DOMDocument.documentElement.childNodes.Item(j -
3).Attributes.Item(1).Text

```

```

sqlCode = "select * from t_GelenTeklifler where 1=2"
con.BeginTrans
rs.Open sqlCode, con, 1

```

```

sqlCode = "update t_GelenTeklifler set alEmToplami="
sqlCode = sqlCode & alEmToplami & ", kazananFirma='5' where "
sqlCode = sqlCode & "alEmno=" & alEmNo & ""
con.Execute sqlCode
rs.Close
con.CommitTrans
con.Close

```

```

FileName = "d:\inetpub\wwwroot\tez\SiparisS3\Sprs*.xml"
strTimeNow = Left(Replace(CStr(FormatDateTime(Now(), 3)), ":", ""), 6)
Set objFileSys = CreateObject("Scripting.FileSystemObject")
Set fileStream = objFileSys.CreateTextFile(Replace(FileName, "*", strTimeNow), False)
fileStream.WriteLine m_DOMDocument.xml
fileStream.Close

```

```

FileName = "d:\inetpub\wwwroot\tez\Satici3\Siparis\Sprs*.xml"
strTimeNow = Left(Replace(CStr(FormatDateTime(Now(), 3)), ":", ""), 6)
Set objFileSys = CreateObject("Scripting.FileSystemObject")
Set fileStream = objFileSys.CreateTextFile(Replace(FileName, "*", strTimeNow), False)
fileStream.WriteLine m_DOMDocument.xml
fileStream.Close

```

```

FileName = "d:\inetpub\wwwroot\tez\logHKK.txt"
strTimeNow = CStr(FormatDateTime(Now(), 1)) & "/" & CStr(FormatDateTime(Now(),
3))
Set objFileSys = CreateObject("Scripting.FileSystemObject")
Set fileStream = objFileSys.OpenTextFile(FileName, ForAppending)
fileStream.WriteLine strTimeNow & " ==> HKKSprsS3 modülü ile " & alEmNo & "
numarali Alim Emri için Satici3e siparis verildi."
fileStream.Close

```

```
IBTSAppIntegration_ProcessMessage = m_DOMDocument.xml  
Set con = Nothing  
Set rs = Nothing  
Set cmd = Nothing  
Set m_DOMDocument = Nothing  
Set objFileSys = Nothing  
Set fileStream = Nothing
```

Exit Function

hata:

```
MsgBox Err.Description & vbCrLf & "Error: " & Err.Number & " (0x" &  
Hex(Err.Number) & ")", vbCritical, "Error " & Err.Source  
Err.Clear
```

End Function



***HKKTklfIste.vbp***

Implements IBTSAppIntegration

Option Explicit

Private m\_DOMDocument As MSXML2.DOMDocument

\*\*\*\*\*

\*\*\*\*\* Gelen Dokuman: T\_alEm.xml, BizTalk Serverdan alinir.

\*\*\*\*\*

Private Function IBTSAppIntegration\_ProcessMessage(  
ByVal bstrDocument As String) As String

Dim objFileSys, strTimeNow, FileName, fileStream

On Error GoTo hata

Set m\_DOMDocument = New MSXML2.DOMDocument

m\_DOMDocument.async = False

IBTSAppIntegration\_ProcessMessage = ""

m\_DOMDocument.loadXML bstrDocument

If m\_DOMDocument.parseError.errorCode <> 0 Then 'The data was not valid XML

Err.Raise m\_DOMDocument.parseError.errorCode, "HKKTklfIste component", \_

"Gelen dokuman islenemiyor : " + \_

vbCrLf + vbCrLf + m\_DOMDocument.parseError.reason + vbCrLf + " satir no: " + \_

CStr(m\_DOMDocument.parseError.Line) + vbCrLf + "yeri: " + \_

CStr(m\_DOMDocument.parseError.linepos)

End If

FileName = "d:\inetpub\wwwroot\tez\Satici1\Teklif\Tklf\*.xml"

strTimeNow = Left(Replace(CStr(FormatDateTime(Now(), 3)), ":", ""), 6)

Set objFileSys = CreateObject("Scripting.FileSystemObject")

Set fileStream = objFileSys.CreateTextFile(Replace(FileName, "\*", strTimeNow), False)

fileStream.WriteLine m\_DOMDocument.xml

fileStream.Close

FileName = "d:\inetpub\wwwroot\tez\Satici2\Teklif\Tklf\*.xml"

strTimeNow = Left(Replace(CStr(FormatDateTime(Now(), 3)), ":", ""), 6)

Set objFileSys = CreateObject("Scripting.FileSystemObject")

Set fileStream = objFileSys.CreateTextFile(Replace(FileName, "\*", strTimeNow), False)

fileStream.WriteLine m\_DOMDocument.xml

fileStream.Close

FileName = "d:\inetpub\wwwroot\tez\Satici3\Teklif\Tklf\*.xml"

strTimeNow = Left(Replace(CStr(FormatDateTime(Now(), 3)), ":", ""), 6)

Set objFileSys = CreateObject("Scripting.FileSystemObject")

Set fileStream = objFileSys.CreateTextFile(Replace(FileName, "\*", strTimeNow), False)

fileStream.WriteLine m\_DOMDocument.xml

fileStream.Close

```
FileName = "d:\inetpub\wwwroot\tez\logHKK.txt"
strTimeNow = CStr(FormatDateTime(Now(), 1)) & " / " & CStr(FormatDateTime(Now(),
3))
Set objFileSys = CreateObject("Scripting.FileSystemObject")
Set fileStream = objFileSys.OpenTextFile(FileName, ForAppending)
fileStream.WriteLine strTimeNow & " ==> Satıcı 1 den teklif istendi."
fileStream.WriteLine strTimeNow & " ==> Satıcı 2 den teklif istendi."
fileStream.WriteLine strTimeNow & " ==> Satıcı 3 den teklif istendi."
fileStream.Close

IBTSAppIntegration_ProcessMessage = bstrDocument
Set m_DOMDocument = Nothing
Set objFileSys = Nothing
Set fileStream = Nothing
Exit Function

hata:
MsgBox Err.Description & vbCrLf & "Error: " & Err.Number & " (0x" &
Hex(Err.Number) & ")", vbCritical, "Error " & Err.Source
Err.Clear

End Function
```

**TeklifKaydet.vbp**

Option Explicit

Private rs, con, cmd, ConnectionString

Private m\_DOMDocument As MSXML2.DOMDocument

Public Sub s1TeklifKaydet(bstr As String)

TeklifKaydet bstr, "1"

End Sub

Public Sub s2TeklifKaydet(bstr As String)

TeklifKaydet bstr, "2"

End Sub

Public Sub s3TeklifKaydet(bstr As String)

TeklifKaydet bstr, "3"

End Sub

Public Sub TeklifKaydet(bstrDocument As String, firma As String)

Dim j, i, fiyat, hesaplanmisFiyat, alEmNo

Dim kod, satirNo, toplam, sqlCode

Dim objFileSys, strTimeNow, FileName, fileStream

On Error GoTo hata

Set m\_DOMDocument = New MSXML2.DOMDocument

m\_DOMDocument.async = False

ConnectionString = "dsn=HKK;server=STUDY;database=HKK;uid=HKKUser;pwd="

Set con = CreateObject("adodb.connection")

Set rs = CreateObject("adodb.recordset")

con.Open ConnectionString

rs.ActiveConnection = con

rs.CursorLocation = adUseClient

Set cmd = CreateObject("adodb.command")

cmd.ActiveConnection = ConnectionString

m\_DOMDocument.loadXML bstrDocument

If m\_DOMDocument.parseError.errorCode &lt;&gt; 0 Then 'The data was not valid XML

Err.Raise m\_DOMDocument.parseError.errorCode, "HKKTeklifYaz component", \_

"Gelen dokuman islenemiyor :" + \_

vbCrLf + vbCrLf + m\_DOMDocument.parseError.reason + vbCrLf + " satir no: " + \_

CStr(m\_DOMDocument.parseError.Line) + vbCrLf + "yeri: " + \_

CStr(m\_DOMDocument.parseError.linepos)

End If

j = m\_DOMDocument.documentElement.childNodes.length

i = 2

alEmNo =

m\_DOMDocument.documentElement.childNodes.Item(0).Attributes.Item(1).Text

sqlCode = "select \* from t\_GlnTkIfAyrinti where 1=2"

con.BeginTrans

rs.Open sqlCode, con, 1

While i &lt; (j - 1)

kod = m\_DOMDocument.documentElement.childNodes.Item(i).Attributes.Item(1).Text

```

    satirNo =
m_DOMDocument.documentElement.childNodes.Item(i).Attributes.Item(6).Text
    fiyat = m_DOMDocument.documentElement.childNodes.Item(i).Attributes.Item(4).Text
    hesaplanmisFiyat =
m_DOMDocument.documentElement.childNodes.Item(i).Attributes.Item(5).Text
    sqlCode = "update t_GlnTklfAyrinti set s" & firma & "Fiyat="
    sqlCode = sqlCode & fiyat & ", s" & firma & "HesaplanmisFiyat=" & hesaplanmisFiyat
    sqlCode = sqlCode & " where alEmNo=" & alEmNo & " and UniversalKod="
    sqlCode = sqlCode & kod & " and satirNo=" & satirNo & ""
    con.Execute sqlCode
    i = i + 1
Wend
rs.Close

```

```

sqlCode = "select * from t_gelenTeklifler where l=2"
rs.Open sqlCode, con, 1
toplam = m_DOMDocument.documentElement.childNodes.Item(j -
1).Attributes.Item(1).Text
sqlCode = "update t_gelenTeklifler set S" & firma & "Toplam=" & toplam
sqlCode = sqlCode & " where alEmNo=" & alEmNo & ""
con.Execute sqlCode
rs.Close
con.CommitTrans
con.Close

```

```

FileName = "d:\inetpub\wwwroot\tez\logHKK.txt"
strTimeNow = CStr(FormatDateTime(Now(), 1)) & " / " & CStr(FormatDateTime(Now(),
3))
Set objFileSys = CreateObject("Scripting.FileSystemObject")
Set fileStream = objFileSys.OpenTextFile(FileName, ForAppending)
fileStream.WriteLine strTimeNow & " ==> Satici" & firma & " teklifi, HKK veritabanina
kaydedildi."
fileStream.Close

```

```

Set m_DOMDocument = Nothing
Set con = Nothing
Set rs = Nothing
Set objFileSys = Nothing
Set fileStream = Nothing
Set cmd = Nothing
Exit Sub

```

```

hata:
MsgBox Err.Description & vbCrLf & "Error: " & Err.Number & " (0x" &
Hex(Err.Number) & ")", vbCritical, "Error " & Err.Source
Err.Clear
End Sub

```



## Ek 6 Satıcı1 için ASP Kodları

### Default.asp

```

<html>
<head>
  <Title> SATICI1</Title>
  <STYLE>
    body {font-family:verdana,helvetica;font-size:8pt}
    button {background-color:"DARKBLUE";font-
family:verdana,helvetica;color="WHITE";font-size:8pt;width:70}
    input {font-family:verdana,helvetica;font-size:8pt}
    Select {font-family:verdana,helvetica;font-size:8pt; valign:top;}
    .Table {background:black}
    .TableHead { font:bold; color:white; background:black}
    .TableColumnHead { font:normal 'Verdana' bold; color:white;background:DARKBLUE}
    .TableRow {font:x-small 'Verdana'; color:black;background:#FAF3FF}
  </STYLE>
</head>

<body>
<table border="0" cellspacing="0" cellpadding="20" height="200" width="95%"
class="Table">
<tr class="TableRow">
  <td width="10%" bgcolor="WHITE">
    &nbsp;
  </td>
  <td bgcolor="#f0E2FC">
    <a href="default.asp">Sat&#305;c&#305;1 </a>
  </td>
  <td bgcolor="#e6CEf8">
    <a href="mal.asp"> Mal </a>
  </td>
  <td bgcolor="#DCBAF4">
    <a href="teklif.asp"> Teklif </a>
  </td>
  <td bgcolor="#d2a6f0">
    <a href="siparis.asp"> Sipari&#351;</a>
  </td>
  <td width="10%" bgcolor="WHITE">
    &nbsp;
  </td>
</tr>
<tr class="TableRow">
<td colspan="8" height="175" bgcolor="#f0E2FC" valign="TOP" >
<font face="Arial" SIZE="2">
<P><BR> SATICI1 sayfas&#305;na HO&#350; GELD&#304;N&#304;Z! <BR>
<BR><br><br>
Bu sayfa ile Mal, Teklif ve Sipari&#351; bilgilerini görebilirsiniz.</P>
</font></td>
</tr></table></body></html>

```

*Mal.asp*

```

<%@ Language=VBScript %>
<html>
<head>
  <Title> SATICI1 - Mal Bilgileri</Title>
  <STYLE>
    body {font-family:verdana,helvetica;font-size:8pt}
    button {background-color:"DARKBLUE";font-
family:verdana,helvetica;color="WHITE";font-size:8pt;width:70}
    input {font-family:verdana,helvetica;font-size:8pt}
    Select {font-family:verdana,helvetica;font-size:8pt; valign:top;}
    .Table {background:black}
    .TableHead { font:bold; color:white; background:black}
    .TableColumnHead { font:normal 'Verdana' bold; color:white;background:DARKBLUE}
    .TableRow {font:x-small 'Verdana'; color:black;background:#FAF3FF}
  </STYLE>

</head>

<BODY LANGUAGE=javascript onerror="return window_onerror()">

<table border="0" cellspacing="0" cellpadding="20" height="200" width="95%"
class="Table">
<tr class="TableRow">
  <td width="10%" bgcolor="WHITE">
    &nbsp;
  </td>
  <td bgcolor="#f0E2FC">
    <a href="default.asp">Sat&#305;c&#305;1 </a>
  </td>
  <td bgcolor="#e6CEf8">
    <a href="mal.asp"> Mal </a>
  </td>
  <td bgcolor="#DCBAF4">
    <a href="teklif.asp"> Teklif </a>
  </td>
  <td bgcolor="#d2a6f0">
    <a href="siparis.asp"> Sipari&#351;</a>
  </td>
  <td width="10%" bgcolor="WHITE">
    &nbsp;
  </td>
</tr>
<tr class="TableRow">
<td colspan="8" height="175" bgcolor="#e6CEf8" valign="TOP" >
<font face="Arial" SIZE="2">

<H1 align=center>Mal Bilgileri</H1>

<!-- #Include File="VeritabaninaBaglan.ASP" -->

```

```

<%
Dim i, j

Client= Request.ServerVariables("HTTP_USER_AGENT")
ClientVer=Split(Client, ";")
rs.Open ("select * from t_mal"), con, 1%>

```

```

<TABLE CLASS="Table" align="center" border = 1>
  <Tr class="TableHead" align="center">
    <td class="TableColumnHead">Mal No</td>
    <td class="TableColumnHead">Universal Kod</td>
    <td class="TableColumnHead">Tan&#305;m</td>
    <td class="TableColumnHead">Ölçü Birimi</td>
    <td class="TableColumnHead">Birim Fiyat</td>
  </tr>

```

```

  <%
    Do while (Not rs.eof) %>

      <tr class="TableRow">
        <% For Index=0 to (rs.fields.count-1) %>
          <TD VAlign=top><% = rs(Index)%></TD>
        <% Next %>
      </tr>

      <% rs.MoveNext
    Loop
  %>

```

```

</TABLE>
<% rs.close
con.close %>

```

```

</td>
</tr>
</table>
</body>
</html>

```

*Siparis.asp*

```

<%@ Language=VBScript %>
<html>
<head>
  <Title> SATICI1 - Sipari&#351; Bilgileri</Title>

  <STYLE>
    body {font-family:verdana,helvetica;font-size:8pt}
    button {background-color:"DARKBLUE";font-
family:verdana,helvetica;color="WHITE";font-size:8pt;width:70}
    input {font-family:verdana,helvetica;font-size:8pt}
    Select {font-family:verdana,helvetica;font-size:8pt; valign:top;}
    .Table {background:black}
    .TableHead { font:bold; color:white; background:black}
    .TableColumnHead { font:normal 'Verdana' bold; color:white;background:DARKBLUE}
    .TableRow {font:x-small 'Verdana'; color:black;background:#FAF3FF}
  </STYLE>

</head>

<BODY LANGUAGE=javascript onerror="return window_onerror()">

<FORM METHOD="post" id=frmSiparis name=frmSiparis>

<table border="0" cellspacing="0" cellpadding="20" height="200" width="95%"
class="Table">
<tr class="TableRow">
  <td width="10%" bgcolor="WHITE">
    &nbsp;
  </td>
  <td bgcolor="#f0E2FC">
    <a href="default.asp">Sat&#305;c&#305;1 </a>
  </td>
  <td bgcolor="#e6CEf8">
    <a href="mal.asp"> Mal </a>
  </td>
  <td bgcolor="#DCBAF4">
    <a href="teklif.asp"> Teklif </a>
  </td>
  <td bgcolor="#d2a6f0">
    <a href="siparis.asp"> Sipari&#351;</a>
  </td>
  <td width="10%" bgcolor="WHITE">
    &nbsp;
  </td>
</tr>
<tr class="TableRow">
<td colspan="8" height="175" bgcolor="#d2a6f0" valign="TOP" >
<font face="Arial" SIZE="2">

```

```
<H1 align=center>Sipari&#351; Bilgileri</H1>
```

```
<!-- #Include File="VeritabaninaBaglan.ASP" -->
```

```
<%
```

```
Dim i, j, rs2, sqlCode, firma, calisan
```

```
Client= Request.ServerVariables("HTTP_USER_AGENT")
```

```
ClientVer=Split(Client, ";")
```

```
set rs2=server.CreateObject("adodb.recordset")
```

```
sqlCode="select * from t_Teklif where siparis='E' order by teklifNo"
```

```
rs.Open (sqlCode), con, 1%>
```

```
<TABLE CLASS="Table" align="center" border = 1>
```

```
<Tr class="TableHead" align="center">
```

```
<td class="TableColumnHead">Teklif No</td>
```

```
<td class="TableColumnHead">Al&#305;m Emri No </td>
```

```
<td class="TableColumnHead">Tarih</td>
```

```
<td class="TableColumnHead">Çal&#305;&#351;an</td>
```

```
<td class="TableColumnHead">Kaç Kalem Mal</td>
```

```
<td class="TableColumnHead">Teklif Toplam&#305;</td>
```

```
<td class="TableColumnHead">Sipari&#351; Veren Kurum</td>
```

```
</tr>
```

```
<%
```

```
Do while (Not rs.eof) %>
```

```
<tr class="TableRow">
```

```
<TD VAlign=top>
```

```
<p align="right"><% = rs("TeklifNo")%></TD>
```

```
<TD VAlign=top>
```

```
<p align="right"><% = rs("AlEmNo")%></TD>
```

```
<TD VAlign=top>
```

```
<p align="center"><% = rs("AlEmTarih")%></TD>
```

```
<TD VAlign=top><% = rs("Calisan")%></TD>
```

```
<TD VAlign=top>
```

```
<p align="center"><% = rs("kacKalemMal")%></TD>
```

```
<TD VAlign=top>
```

```
<p align="right"><% = rs("alEmToplami")%></TD>
```

```
<%
```

```
firma=rs("teklifIsteyenFirma")
```

```
rs2.Open ("select ad from t_kurum where kurumNo= '' & firma & ''"), con, 1
```

```
%>
```

```
<TD VAlign=top>
```

```
<p align="center"><% = trim(rs2("ad"))%></TD>
```

```
</tr>
```

```
<% rs2.close
```

```
rs.MoveNext
```

```
Loop
```

```
%>
```

```

</TABLE>
<% rs.close
con.close
set rs=Nothing
set rs2=Nothing
set con=Nothing %>
<br>
<br>
<table border="0" align="center">
<tr>
<td width="240"> Ayr&#305;nt&#305;s&#305;n&#305; görmek istedi&#287;iniz
TeklifNo :</td>
<td><input type="text" name="txtTeklifNo" size="5" value=" "></td>
</tr>
</table>
<table border="0" align="center">
<tr>
<td><button onclick="Ayrinti()">Ayr&#305;nt&#305;</button></td>
<td><button onclick="window.close()">Kapat</button></td>
</tr>
</TABLE>

</td></tr>
</table>
<script language="Javascript">
var mywin
var prevdiv="dummydiv"

function Ayrinti()
{
var str, i;

i=frmSiparis.elements('txtTeklifNo').value ;

str="siparis2.asp?TeklifNo="+ i;

if (mywin!=null)
{
if (!mywin.closed)
{
mywin.close();
mywin=null;
}
}
mywin=window.open(str,null,"directories=no,location=no,menubar=no,status,toolbar=no,resizeable,scrollbars");
window.event.cancelBubble=true;
}
</script>
</html>

```

*Siparis2.asp*

```

<%@ Language=VBScript %>
<HTML>
  <HEAD> <TITLE>SATICI1 - Sipari&#351; Bilgileri </TITLE>

  <STYLE>
    body {font-family:verdana,helvetica;font-size:8pt}
    button {background-color:"DARKBLUE";font-
family:verdana,helvetica;color="WHITE";font-size:8pt;width:70}
    input {font-family:verdana,helvetica;font-size:8pt}
    Select {font-family:verdana,helvetica;font-size:8pt; valign:top;}
    .Table {background:black}
    .TableHead { font:bold; color:white; background:black}
    .TableColumnHead { font:normal 'Verdana' bold; color:white;background:DARKBLUE}
    .TableRow {font:x-small 'Verdana'; color:black;background:#FAF3FF}
  </STYLE>

</HEAD>

<BODY bgcolor="#d2a6f0" LANGUAGE=javascript onerror="return window_onerror()">

<!-- #Include File="VeritabaninaBaglan.ASP" -->

<FORM METHOD="post" id=frmTklf2 name=frmTklf2>
<%
  Dim teklifNo, rs2, sqlCode

  teklifNo=Request.QueryString("TeklifNo")
  Client= Request.ServerVariables("HTTP_USER_AGENT")
  ClientVer=Split(Client, ";")
  sqlCode="select * from t_teklif where siparis = 'E' and TeklifNo="" & teklifNo & """"
  rs.Open (sqlcode), con, 1
  if not (rs.eof) then
    rs.close
    sqlCode="select * from t_teklifAyrinti where teklifNo = "" & teklifNo & "" order by
satirNo"
    rs.Open (sqlcode), con, 1
    set rs2=server.CreateObject("adodb.recordset")

  %>
  <TABLE CLASS="Table" align="center" border = 1>
    <Tr class="TableHead" align="center">
      <td class="TableColumnHead">Mal Tan&#305;m&#305;</td>
      <td class="TableColumnHead">Adet</td>
      <td class="TableColumnHead">Fiyat</td>
      <td class="TableColumnHead">Hesaplanm&#305;&#351; Fiyat</td>
    </tr>

```

```

<%
  Do while (Not rs.eof) %>
    <tr class="TableRow">
      <%rs2.Open ("select tanim from t_mal where universalKod = "" &
rs("universalKod") & """), con, 1 %>
      <TD VAlign=top><% = rs2("tanim")%></TD>
      <TD VAlign=top>
        <p align="center"><% = rs("adet")%></TD>
      <TD VAlign=top>
        <p align="right"><% = rs("fiyat")%></TD>
      <TD VAlign=top>
        <p align="right"><% = rs("hesaplanmisFiyat")%></TD>
    </tr>
    <% rs2.close
      rs.MoveNext
    Loop
  %>
</TABLE>
<%
  else response.write teklifNo & " numaral&#305; sipari&#351; yoktur."
  end if
  rs.close
  con.close
  set rs=Nothing
  set con=Nothing %>
<br><br>
<center><button onclick="window.close()">Kapat</button></center>
</FORM>&nbsp;
</BODY>
</HTML>

```



*Teklif.asp*

```

<%@ Language=VBScript %>
<html>
<head>
  <Title> SATICI1 - Teklif Bilgileri</Title>

  <STYLE>
    body {font-family:verdana,helvetica;font-size:8pt}
    button {background-color:"DARKBLUE";font-
family:verdana,helvetica;color="WHITE";font-size:8pt;width:70}
    input {font-family:verdana,helvetica;font-size:8pt}
    Select {font-family:verdana,helvetica;font-size:8pt; valign:top;}
    .Table {background:black}
    .TableHead { font:bold; color:white; background:black}
    .TableColumnHead { font:normal 'Verdana' bold; color:white;background:DARKBLUE}
    .TableRow {font:x-small 'Verdana'; color:black;background:#FAF3FF}
  </STYLE>

</head>

<BODY LANGUAGE=javascript onerror="return window_onerror()">

<FORM METHOD="post" id=frmTeklif name=frmTeklif>

<table border="0" cellspacing="0" cellpadding="20" height="200" width="95%"
class="Table">
<tr class="TableRow">
  <td width="10%" bgcolor="WHITE">
    &nbsp;
  </td>
  <td bgcolor="#f0E2FC">
    <a href="default.asp">Sat&#305;c&#305;l </a>
  </td>
  <td bgcolor="#e6CEf8">
    <a href="mal.asp"> Mal </a>
  </td>
  <td bgcolor="#DCBAF4">
    <a href="teklif.asp"> Teklif </a>
  </td>
  <td bgcolor="#d2a6f0">
    <a href="siparis.asp"> Sipari&#351;</a>
  </td>
  <td width="10%" bgcolor="WHITE">
    &nbsp;
  </td>
</tr>
<tr class="TableRow">
<td colspan="8" height="175" bgcolor="#DCBAF4" valign="TOP" >
<font face="Arial" SIZE="2">

```

```
<H1 align=center>Teklif Bilgileri</H1>
```

```
<!-- #Include File="VeritabaninaBaglan.ASP" -->
```

```
<%
```

```
Dim i, j, rs2, sqlCode, firma, calisan
```

```
Client= Request.ServerVariables("HTTP_USER_AGENT")
```

```
ClientVer=Split(Client,",")
```

```
set rs2=server.CreateObject("adodb.recordset")
```

```
sqlCode="select * from t_Teklif where siparis='H' order by teklifNo"
```

```
rs.Open (sqlCode), con, 1%>
```

```
<TABLE CLASS="Table" align="center" border = 1>
```

```
<Tr class="TableHead" align="center">
```

```
<td class="TableColumnHead">Teklif No</td>
```

```
<td class="TableColumnHead">Al&#305;m Emri No </td>
```

```
<td class="TableColumnHead">Tarih</td>
```

```
<td class="TableColumnHead">Çal&#305;,&#351;an</td>
```

```
<td class="TableColumnHead">Kaç Kalem Mal</td>
```

```
<td class="TableColumnHead">Teklif Toplam&#305;</td>
```

```
<td class="TableColumnHead">Teklif &#304;steyen Kurum</td>
```

```
</tr>
```

```
<%
```

```
Do while (Not rs.eof) %>
```

```
<tr class="TableRow">
```

```
<TD VAlign=top>
```

```
<p align="right"><% = rs("TeklifNo")%></TD>
```

```
<TD VAlign=top>
```

```
<p align="right"><% = rs("AlEmNo")%></TD>
```

```
<TD VAlign=top>
```

```
<p align="center"><% = rs("AlEmTarih")%></TD>
```

```
<TD VAlign=top><% = rs("Calisan")%></TD>
```

```
<TD VAlign=top>
```

```
<p align="center"><% = rs("kacKalemMal")%></TD>
```

```
<TD VAlign=top>
```

```
<p align="right"><% = rs("alEmToplami")%></TD>
```

```
<%
```

```
firma=rs("teklifsteyenFirma")
```

```
rs2.Open ("select ad from t_kurum where kurumNo= "" & firma & """), con, 1
```

```
%>
```

```
<TD VAlign=top>
```

```
<p align="center"><% = trim(rs2("ad"))%></TD>
```

```
</tr>
```

```
<% rs2.close
```

```
rs.MoveNext
```

```
Loop
```

```
%>
```

```
</TABLE>
```

```

<% rs.close
con.close
set rs=Nothing
set rs2=Nothing
set con=Nothing %>
<br>
<br>
<table border="0" align="center">
<tr>
<td width="240"> Ayr&#305;nt&#305;s&#305;n&#305; görmek istedi&#287;iniz
TeklifNo :</td>
<td><input type="text" name="txtTeklifNo" size="5" value=" "></td>
</tr>
</table>
<table border="0" align="center">
<tr>
<td><button onclick="Ayrinti()">Ayr&#305;nt&#305;</button></td>
<td><button onclick="window.close()">Kapat</button></td>
</tr>
</TABLE>

</td>
</tr>
</table>

<script language="Javascript">
var mywin
var prevdiv="dummydiv"

function Ayrinti()
{
var str, i;

i=frmTeklif.elements('txtTeklifNo').value ;

str="teklif2.asp?TeklifNo="+ i;

if (mywin!=null)
{
if (!mywin.closed)
{
mywin.close();
mywin=null;
}
}
mywin=window.open(str,null,"directories=no,location=no,menubar=no,status,toolbar=no,r
esizable,scrollbars");
window.event.cancelBubble=true;
}</script>
</html>

```

**Teklif2.asp**

```

<%@ Language=VBScript %>
<HTML>
  <HEAD> <TITLE>SATICI1 - Teklif Bilgileri </TITLE>

  <STYLE>
    body {font-family:verdana,helvetica;font-size:8pt}
    button {background-color:"DARKBLUE";font-
family:verdana,helvetica;color="WHITE";font-size:8pt;width:70}
    input {font-family:verdana,helvetica;font-size:8pt}
    Select {font-family:verdana,helvetica;font-size:8pt; valign:top;}
    .Table {background:black}
    .TableHead { font:bold; color:white; background:black}
    .TableColumnHead { font:normal 'Verdana' bold; color:white;background:DARKBLUE}
    .TableRow {font:x-small 'Verdana'; color:black;background:#FAF3FF}
  </STYLE>

</HEAD>

<BODY bgcolor="#DCBAF4" LANGUAGE=javascript onerror="return window_onerror()">

<!-- #Include File="VeritabaninaBaglan.ASP" -->

<FORM METHOD="post" id=frmTklf2 name=frmTklf2>
<%

Dim teklifNo, rs2, sqlCode

teklifNo=Request.QueryString("TeklifNo")
Client= Request.ServerVariables("HTTP_USER_AGENT")
ClientVer=Split(Client,",")

sqlCode="select * from t_teklif where siparis = 'H' and TeklifNo=" & teklifNo & ""
rs.Open (sqlcode), con, 1
if not (rs.eof) then
  rs.close
  sqlCode="select * from t_teklifAyrinti where teklifNo = " & teklifNo & "" order by
satirNo"
  rs.Open (sqlcode), con, 1
  set rs2=server.CreateObject("adodb.recordset")

%>
<TABLE CLASS="Table" align="center" border = 1>
  <Tr class="TableHead" align="center">
    <td class="TableColumnHead">Mal Tan&#305;m&#305;</td>
    <td class="TableColumnHead">Adet</td>
    <td class="TableColumnHead">Fiyat</td>
    <td class="TableColumnHead">Hesaplanm&#305;&#351; Fiyat</td>
  </tr>

```

```

<%
  Do while (Not rs.eof) %>
    <tr class="TableRow">
      <%rs2.Open ("select tanim from t_mal where universalKod = "" &
rs("universalKod") & """), con, 1 %>
      <TD VAlign=top><% = rs2("tanim")%></TD>
      <TD VAlign=top>
        <p align="center"><% = rs("adet")%></TD>
      <TD VAlign=top>
        <p align="right"><% = rs("fiyat")%></TD>
      <TD VAlign=top>
        <p align="right"><% = rs("hesaplanmisFiyat")%></TD>
    </tr>
    <% rs2.close
      rs.MoveNext
  Loop
%>
</TABLE>
<% else response.write teklifNo & " numaral&#305; teklif yoktur."
  end if
  rs.close
  con.close
  set rs=Nothing
  set con=Nothing %>
<br><br>
<center><button onclick="window.close()">Kapat</button></center>
</FORM>&nbsp;
</BODY>
</HTML>

```

***VeritabaninaBaglan.asp***

```
<!-- #Include File="ADOVBS.INC" -->
<%
Dim rs ,con, ConnectionString

ConnectionString = "dsn=Satici1;server=STUDY;database=Satici1;uid=Satici1 User;pwd="
set con=server.CreateObject ("adodb.connection")
set rs=server.CreateObject("adodb.recordset")
con.Open connectionstring
Rs.ActiveConnection = Con
Rs.CursorLocation = adUseClient
set Cmd= server.CreateObject("adodb.command")
Cmd.ActiveConnection =ConnectionString
%>
```



## Ek 7 Satıcı1 için COM Kodları

### *SISprsHKKAIC.vbp*

Implements IBTSAppIntegration

Option Explicit

Private rs, con, cmd, ConnectionString

Private m\_DOMDocument As MSXML2.DOMDocument

Private m\_objSubmit As BTSInterchangeLib.Interchange

\*\*\*\*\*

\*\*\*\*\* Gelen Dokuman: T\_Fatura, BizTalk Serverdan alinir.

\*\*\*\*\*

Private Function IBTSAppIntegration\_ProcessMessage(  
ByVal bstrDocument As String) As String

Dim teklifNo, j, alEmToplami, tarih, kurumNo, gun, ay, yil

Dim objFileSys, strTimeNow, FileName, FileStream

Dim alEmNo, sqlCode

On Error GoTo hata

ConnectionString = "dsn=Satıcı1;server=STUDY;database=Satıcı1;uid=Satıcı1 User;pwd="

Set con = CreateObject("adodb.connection")

Set rs = CreateObject("adodb.recordset")

con.open ConnectionString

rs.ActiveConnection = con

rs.CursorLocation = adUseClient

Set cmd = CreateObject("adodb.command")

cmd.ActiveConnection = ConnectionString

Set m\_DOMDocument = New MSXML2.DOMDocument

m\_DOMDocument.async = False

Set m\_objSubmit = New BTSInterchangeLib.Interchange

IBTSAppIntegration\_ProcessMessage = ""

m\_DOMDocument.loadXML bstrDocument

If m\_DOMDocument.parseError.errorCode <> 0 Then 'The data was not valid XML

Err.Raise m\_DOMDocument.parseError.errorCode, "Satıcı1 Siparis component", \_

"Gelen dokuman islenemiyor : " + \_

vbCrLf + vbCrLf + m\_DOMDocument.parseError.reason + vbCrLf + " satir no: " + \_

CStr(m\_DOMDocument.parseError.Line) + vbCrLf + "yeri: " + \_

CStr(m\_DOMDocument.parseError.linepos)

End If

alEmNo =

m\_DOMDocument.documentElement.childNodes.Item(0).Attributes.Item(2).Text

con.BeginTrans

rs.open ("select \* from t\_teklif where alEmNo=" & alEmNo & """), con, 1

```

teklifNo = Trim(rs!teklifNo)
kurumNo = Trim(rs!teklifIsteyenFirma)
con.Execute ("update t_teklif set siparis='E' where alEmNo='" & alEmNo & "'")
rs.Close
con.CommitTrans
con.Close

```

```

j = m_DOMDocument.documentElement.childNodes.length

```

```

gun = Day(Now())
ay = Month(Now())
yil = Year(Now())
If (Len(gun) = 1) Then
    gun = "0" & gun
End If
If (Len(ay) = 1) Then
    ay = "0" & ay
End If
tarih = yil & "-" & ay & "-" & gun

```

```

m_DOMDocument.documentElement.childNodes.Item(0).Attributes.Item(0).Text = tarih
m_DOMDocument.documentElement.childNodes.Item(0).Attributes.Item(1).Text =
teklifNo
m_DOMDocument.documentElement.childNodes.Item(0).Attributes.Item(4).Text = tarih

```

```

m_DOMDocument.documentElement.childNodes.Item(1).childNodes.Item(0).Attributes.Item
(0).Text = "Satici1"

```

```

m_DOMDocument.documentElement.childNodes.Item(1).childNodes.Item(0).Attributes.Item
(1).Text = "Merkez"

```

```

m_DOMDocument.documentElement.childNodes.Item(1).childNodes.Item(0).Attributes.Item
(3).Text = "IZMIR"

```

```

m_DOMDocument.documentElement.childNodes.Item(1).childNodes.Item(0).Attributes.Item
(4).Text = "35000"

```

```

m_DOMDocument.documentElement.childNodes.Item(1).childNodes.Item(1).Attributes.Item
(0).Text = "Satici1 Calisani"

```

```

m_DOMDocument.documentElement.childNodes.Item(1).childNodes.Item(1).Attributes.Item
(1).Text = "1245"

```

```

m_DOMDocument.documentElement.childNodes.Item(2).childNodes.Item(0).Attributes.Item
(0).Text = "Hava Kuvvetleri K.ligi"

```

```

m_DOMDocument.documentElement.childNodes.Item(2).childNodes.Item(0).Attributes.Item
(1).Text = "Merkez"

```



```
m_DOMDocument.documentElement.childNodes.Item(2).childNodes.Item(0).Attributes.Item(3).Text = "ANKARA"
```

```
m_DOMDocument.documentElement.childNodes.Item(2).childNodes.Item(0).Attributes.Item(4).Text = "06000"
```

```
m_DOMDocument.documentElement.childNodes.Item(3).Attributes.Item(0).Text = "10"
```

```
alEmToplami = m_DOMDocument.documentElement.childNodes.Item(j - 3).Attributes.Item(1).Text
```

```
m_DOMDocument.documentElement.childNodes.Item(j - 3).Attributes.Item(2).Text = Round(alEmToplami * 0.05)
```

```
m_DOMDocument.documentElement.childNodes.Item(j - 3).Attributes.Item(3).Text = Round(alEmToplami * 0.1)
```

```
m_DOMDocument.documentElement.childNodes.Item(j - 3).Attributes.Item(4).Text = Round(alEmToplami -
```

```
    m_DOMDocument.documentElement.childNodes.Item(j - 3).Attributes.Item(2).Text +
    m_DOMDocument.documentElement.childNodes.Item(j - 3).Attributes.Item(3).Text)
```

```
FileName = "d:\inetpub\wwwroot\tez\Satici1\logSatici1.txt"
```

```
strTimeNow = CStr(FormatDateTime(Now(), 1)) & " / " & CStr(FormatDateTime(Now(), 3))
```

```
Set objFileSys = CreateObject("Scripting.FileSystemObject")
```

```
Set fileStream = objFileSys.OpenTextFile(FileName, ForAppending)
```

```
fileStream.WriteLine strTimeNow & " ==> Satici1 fatura gönderdi."
```

```
fileStream.Close
```

```
Set objFileSys = Nothing
```

```
Set fileStream = Nothing
```

```
IBTSAppIntegration_ProcessMessage =
```

```
m_objSubmit.Submit(BIZTALK_OPENNESS_TYPE_NOTOPEN, m_DOMDocument.xml, , "T_Satici1 on Seller", , "T_Satici1 on Buyer(HKK)")
```

```
Set con = Nothing
```

```
Set rs = Nothing
```

```
Set cmd = Nothing
```

```
Set m_DOMDocument = Nothing
```

```
Set m_objSubmit = Nothing
```

```
Exit Function
```

```
hata:
```

```
MsgBox Err.Description & vbCrLf & "Error: " & Err.Number & " (0x" & Hex(Err.Number) & ")", vbCritical, "Error " & Err.Source
```

```
Err.Clear
```

```
End Function
```

**S1SprsHKKQ.vbp**

Implements IBTSAppIntegration

Option Explicit

Private g\_MSMTxDisp As MSMQ.MSMQTransactionDispenser

Private g\_MSMQQueue As MSMQ.MSMQQueue

Private g\_MSMQInfo As MSMQ.MSMQQueueInfo

Private m\_DOMDocument As MSXML2.DOMDocument

Private g\_SkedFile As String

\*\*\*\*\*

\*\*\*\*\* Gelen Dokuman: T\_alEm.xml, BizTalk Serverdan alinir.

\*\*\*\*\*

Private Function IBTSAppIntegration\_ProcessMessage(  
ByVal bstrDocument As String) As String

Dim objFileSys, strTimeNow, FileName, fileStream

On Error GoTo hata

Set m\_DOMDocument = New MSXML2.DOMDocument

m\_DOMDocument.async = False

IBTSAppIntegration\_ProcessMessage = ""

Set g\_MSMQInfo = CreateObject("MSMQ.MSMQQueueInfo")

Set g\_MSMTxDisp = CreateObject("MSMQ.MSMQTransactionDispenser")

m\_DOMDocument.loadXML bstrDocument

If m\_DOMDocument.parseError.errorCode <> 0 Then "The data was not valid XML

Err.Raise m\_DOMDocument.parseError.errorCode, "S1SprsHKKQ component", \_  
"Gelen dokuman islenemiyor :" + \_  
vbCrLf + vbCrLf + m\_DOMDocument.parseError.reason + vbCrLf + " satir no: " + \_  
CStr(m\_DOMDocument.parseError.Line) + vbCrLf + "yeri: " + \_  
CStr(m\_DOMDocument.parseError.linepos)

End If

'satici1 schedule

g\_SkedFile = "sked:/// " & "d:\inetpub\wwwroot\tez\Satici1\Schedule\S1HKKSiiparis.skx"

'Connect To MSMQ and Remove Any Existing Messages

PurgeMSMQ "DIRECT=OS:\private\$\S1Siiparis"

'Send Selected message to MSMQ

ExecuteMSMQ "DIRECT=OS:\private\$\S1Siiparis", bstrDocument

ExecuteSchedule g\_SkedFile

Set objFileSys = Nothing

FileName = "d:\inetpub\wwwroot\tez\Satici1\logSatici1.txt"

strTimeNow = CStr(FormatDateTime(Now(), 1)) & " / " & CStr(FormatDateTime(Now(),

3))

Set objFileSys = CreateObject("Scripting.FileSystemObject")

Set fileStream = objFileSys.OpenTextFile(FileName, ForAppending)

fileStream.WriteLine strTimeNow & " ==> S1Siiparis kuyruguna HKK dokumani atandi."

fileStream.WriteLine strTimeNow & " ==> Satici1 SiiparisHKK Schedule aktif edildi."

```
fileStream.Close
```

```
IBTSAppIntegration_ProcessMessage = bstrDocument
```

```
Set objFileSys = Nothing
Set fileStream = Nothing
Set g_MSMQInfo = Nothing
Set g_MSMTxDisp = Nothing
Set m_DOMDocument = Nothing
Exit Function
```

```
hata:
```

```
MsgBox Err.Description & vbCrLf & "Error: " & Err.Number & " (0x" &
Hex(Err.Number) & ")", vbCritical, "Error " & Err.Source
Err.Clear
```

```
End Function
```

```
Public Sub ExecuteMSMQ(ByVal strQueuePath As String, DataToQueue As String)
```

```
Dim QueueMsg As New MSMQMessage
Dim MSMQTx
Dim strData As String
Dim fSend As Boolean
Dim txt As TextStream
Dim mybyte() As Byte
```

```
On Error GoTo Err_SendMSMQ
g_MSMQInfo.FormatName = strQueuePath
Set g_MSMQQueue = g_MSMQInfo.Open(MQ_SEND_ACCESS, MQ_DENY_NONE)
mybyte = StrConv(DataToQueue, vbFromUnicode)
QueueMsg.Body = DataToQueue
```

```
Set MSMQTx = g_MSMTxDisp.BeginTransaction
QueueMsg.Send g_MSMQQueue, MSMQTx
MSMQTx.Commit
```

```
Set QueueMsg = Nothing
Set MSMQTx = Nothing
Exit Sub
```

```
Err_SendMSMQ:
```

```
Err.Raise Err.Number, "Kuyruga mesaji gonderirken", "MSMQ'ya mesaj gonderilemiyor
"" & strQueuePath & ""." & vbCrLf & vbCrLf & Err.Description
Exit Sub
End Sub
```

```
Public Sub PurgeMSMQ(ByVal strQueuePath As String)
```

```
Dim l_MSMQMsg As MSMQMessage

On Error GoTo Err_ConnectMSMQ
g_MSMQInfo.FormatName = strQueuePath
```

```
Set g_MSMQQueue = g_MSMQInfo.Open(MQ_RECEIVE_ACCESS,
MQ_DENY_NONE)
```

```
On Error GoTo Err_PurgeMSMQ
Do
  Set l_MSMQMsg = g_MSMQQueue.Receive(, , 1)
Loop While Not l_MSMQMsg Is Nothing
```

```
Exit Sub
```

```
Err_ConnectMSMQ:
```

```
  Err.Raise Err.Number, "Connecting To MSMQ", "Could Not Open the MSMQ Queue """"
  & strQueuePath & """"." & vbCrLf & vbCrLf & Err.Description
```

```
  Exit Sub
```

```
Err_PurgeMSMQ:
```

```
  Err.Raise Err.Number, "Cleaning MSMQ", "Could Not Remove Existing Messages from
  MSMQ Queue """" & strQueuePath & """"." & vbCrLf & vbCrLf & Err.Description
```

```
  Exit Sub
```

```
End Sub
```

```
Public Sub ExecuteSchedule(ByVal strSchedule)
```

```
  Dim SendPAQ As Object
```

```
  On Error GoTo Err_ExecSched
```

```
  Set SendPAQ = GetObject(strSchedule)
```

```
  If SendPAQ Is Nothing Then
```

```
    Err.Raise vbObjectError + 1, , "Invalid Schedule Handle Returned."
```

```
  End If
```

```
  Set SendPAQ = Nothing
```

```
  Exit Sub
```

```
Err_ExecSched:
```

```
  Err.Raise Err.Number, "Starting Schedule", "Could Not Launch the XLANG Schedule" &
  vbCrLf & "Please verify the path to the SKX file and path to your data are correct. Also
  make sure the private queues have been created." & vbCrLf & vbCrLf & Err.Description
```

```
  Exit Sub
```

```
End Sub
```

*S1SprsKKKAIC.vbp*

Implements IBTSAppIntegration

Option Explicit

Private rs, con, cmd, ConnectionString

Private m\_DOMDocument As MSXML2.DOMDocument

Private m\_objSubmit As BTSInterchangeLib.Interchange

```
*****
***** Gelen Dokuman: T_Fatura, BizTalk Serverdan alinir.
*****
```

Private Function IBTSAppIntegration\_ProcessMessage( \_  
ByVal bstrDocument As String) As String

Dim teklifNo, j, alEmToplami, tarih, kurumNo, gun, ay, yil

Dim objFileSys, strTimeNow, FileName, fileStream

Dim alEmNo, sqlCode

On Error GoTo hata

ConnectionString = "dsn=Satici1;server=STUDY;database=Satici1;uid=Satici1User;pwd="

Set con = CreateObject("adodb.connection")

Set rs = CreateObject("adodb.recordset")

con.open ConnectionString

rs.ActiveConnection = con

rs.CursorLocation = adUseClient

Set cmd = CreateObject("adodb.command")

cmd.ActiveConnection = ConnectionString

Set m\_DOMDocument = New MSXML2.DOMDocument

m\_DOMDocument.async = False

Set m\_objSubmit = New BTSInterchangeLib.Interchange

IBTSAppIntegration\_ProcessMessage = ""

m\_DOMDocument.loadXML bstrDocument

If m\_DOMDocument.parseError.errorCode <> 0 Then 'The data was not valid XML

Err.Raise m\_DOMDocument.parseError.errorCode, "Satici1 Siparis component", \_

"Gelen dokuman islenemiyor : " + \_

vbCrLf + vbCrLf + m\_DOMDocument.parseError.reason + vbCrLf + " satir no: " + \_

CStr(m\_DOMDocument.parseError.Line) + vbCrLf + "yeri: " + \_

CStr(m\_DOMDocument.parseError.linepos)

End If

alEmNo =

m\_DOMDocument.documentElement.childNodes.Item(0).Attributes.Item(2).Text

con.BeginTrans

rs.open ("select \* from t\_teklif where alEmNo='" & alEmNo & "'", con, 1

teklifNo = Trim(rs!teklifNo)

kurumNo = Trim(rs!teklifIsteyenFirma)

```

con.Execute ("update t_teklif set siparis='E' where alEmNo='" & alEmNo & "'")
rs.Close
con.CommitTrans
con.Close

```

```

j = m_DOMDocument.documentElement.childNodes.length

```

```

gun = Day(Now())
ay = Month(Now())
yil = Year(Now())
If (Len(gun) = 1) Then
    gun = "0" & gun
End If
If (Len(ay) = 1) Then
    ay = "0" & ay
End If
tarih = yil & "-" & ay & "-" & gun

```

```

m_DOMDocument.documentElement.childNodes.Item(0).Attributes.Item(0).Text = tarih
m_DOMDocument.documentElement.childNodes.Item(0).Attributes.Item(1).Text =
teklifNo
m_DOMDocument.documentElement.childNodes.Item(0).Attributes.Item(4).Text = tarih

```

```

m_DOMDocument.documentElement.childNodes.Item(1).childNodes.Item(0).Attributes.Item
(0).Text = "Satici1"

```

```

m_DOMDocument.documentElement.childNodes.Item(1).childNodes.Item(0).Attributes.Item
(1).Text = "Merkez"

```

```

m_DOMDocument.documentElement.childNodes.Item(1).childNodes.Item(0).Attributes.Item
(3).Text = "IZMIR"

```

```

m_DOMDocument.documentElement.childNodes.Item(1).childNodes.Item(0).Attributes.Item
(4).Text = "35000"

```

```

m_DOMDocument.documentElement.childNodes.Item(1).childNodes.Item(1).Attributes.Item
(0).Text = "Satici1 Calisani"

```

```

m_DOMDocument.documentElement.childNodes.Item(1).childNodes.Item(1).Attributes.Item
(1).Text = "1245"

```

```

m_DOMDocument.documentElement.childNodes.Item(2).childNodes.Item(0).Attributes.Item
(0).Text = "Kara Kuvvetleri K.ligi"

```

```

m_DOMDocument.documentElement.childNodes.Item(2).childNodes.Item(0).Attributes.Item
(1).Text = "Merkez"

```

```

m_DOMDocument.documentElement.childNodes.Item(2).childNodes.Item(0).Attributes.Item
(3).Text = "ANKARA"

```

```
m_DOMDocument.documentElement.childNodes.Item(2).childNodes.Item(0).Attributes.Item(4).Text = "06000"
```

```
m_DOMDocument.documentElement.childNodes.Item(3).Attributes.Item(0).Text = "10"
```

```
alEmToplami = m_DOMDocument.documentElement.childNodes.Item(j - 3).Attributes.Item(1).Text
```

```
m_DOMDocument.documentElement.childNodes.Item(j - 3).Attributes.Item(2).Text = Round(alEmToplami * 0.05)
```

```
m_DOMDocument.documentElement.childNodes.Item(j - 3).Attributes.Item(3).Text = Round(alEmToplami * 0.1)
```

```
m_DOMDocument.documentElement.childNodes.Item(j - 3).Attributes.Item(4).Text = Round(alEmToplami -
```

```
m_DOMDocument.documentElement.childNodes.Item(j - 3).Attributes.Item(2).Text +
```

```
m_DOMDocument.documentElement.childNodes.Item(j - 3).Attributes.Item(3).Text)
```

```
FileName = "d:\inetpub\wwwroot\tez\Satici1\logSatici1.txt"
```

```
strTimeNow = CStr(FormatDateTime(Now(), 1)) & "/" & CStr(FormatDateTime(Now(), 3))
```

```
Set objFileSys = CreateObject("Scripting.FileSystemObject")
```

```
Set fileStream = objFileSys.OpenTextFile(FileName, ForAppending)
```

```
fileStream.WriteLine strTimeNow & " ==> Satici1 fatura gönderdi."
```

```
fileStream.Close
```

```
Set objFileSys = Nothing
```

```
Set fileStream = Nothing
```

```
IBTSAAppIntegration_ProcessMessage =
```

```
m_objSubmit.Submit(BIZTALK_OPENNESS_TYPE_NOTOPEN, m_DOMDocument.xml, , "T_Satici1 on Seller", , "T_Satici1 on Buyer(KKK)")
```

```
Set con = Nothing
```

```
Set rs = Nothing
```

```
Set cmd = Nothing
```

```
Set m_DOMDocument = Nothing
```

```
Set m_objSubmit = Nothing
```

```
Exit Function
```

```
hata:
```

```
MsgBox Err.Description & vbCrLf & "Error: " & Err.Number & " (0x" & Hex(Err.Number) & ")", vbCritical, "Error " & Err.Source
```

```
Err.Clear
```

```
End Function
```

**S1SprsKKKQ.vbp**

Implements IBTSAppIntegration

Option Explicit

Private g\_MSMTxDisp As MSMQ.MSMQTransactionDispenser

Private g\_MSMQQueue As MSMQ.MSMQQueue

Private g\_MSMQInfo As MSMQ.MSMQQueueInfo

Private m\_DOMDocument As MSXML2.DOMDocument

Private g\_SkedFile As String

\*\*\*\*\*

\*\*\*\*\* Gelen Dokuman: T\_alEm.xml, BizTalk Serverdan alinir.

\*\*\*\*\*

Private Function IBTSAppIntegration\_ProcessMessage(  
ByVal bstrDocument As String) As String

Dim objFileSys, strTimeNow, FileName, fileStream

On Error GoTo hata

Set m\_DOMDocument = New MSXML2.DOMDocument

m\_DOMDocument.async = False

IBTSAppIntegration\_ProcessMessage = ""

Set g\_MSMQInfo = CreateObject("MSMQ.MSMQQueueInfo")

Set g\_MSMTxDisp = CreateObject("MSMQ.MSMQTransactionDispenser")

m\_DOMDocument.loadXML bstrDocument

If m\_DOMDocument.parseError.errorCode <> 0 Then 'The data was not valid XML

Err.Raise m\_DOMDocument.parseError.errorCode, "S1SprsKKKQ component", \_

"Gelen dokuman islenemiyor :" + \_

vbCrLf + vbCrLf + m\_DOMDocument.parseError.reason + vbCrLf + " satir no: " + \_

CStr(m\_DOMDocument.parseError.Line) + vbCrLf + "yeri: " + \_

CStr(m\_DOMDocument.parseError.linepos)

End If

'satici1 schedule

g\_SkedFile = "sked:/// " & "d:\inetpub\wwwroot\tez\Satici1\Schedule\S1KKKSiparis.skx"

'Connect To MSMQ and Remove Any Existing Messages

PurgeMSMQ "DIRECT=OS:\private\$\S1Siparis"

'Send Selected message to MSMQ

ExecuteMSMQ "DIRECT=OS:\private\$\S1Siparis", bstrDocument

ExecuteSchedule g\_SkedFile

Set objFileSys = Nothing

FileName = "d:\inetpub\wwwroot\tez\Satici1\logSatici1.txt"

strTimeNow = CStr(FormatDateTime(Now(), 1)) & " / " & CStr(FormatDateTime(Now(),

3))

Set objFileSys = CreateObject("Scripting.FileSystemObject")

Set fileStream = objFileSys.OpenTextFile(FileName, ForAppending)

fileStream.WriteLine strTimeNow & " ==> S1Siparis kuyruguna KKK dokumani atandi."

fileStream.WriteLine strTimeNow & " ==> Satici1 SiparisKKK Schedule aktif edildi."



```
fileStream.Close
```

```
IBTSAppIntegration_ProcessMessage = bstrDocument
```

```
Set objFileSys = Nothing
Set fileStream = Nothing
Set g_MSMQInfo = Nothing
Set g_MSMTxDisp = Nothing
Set m_DOMDocument = Nothing
Exit Function
```

```
hata:
```

```
MsgBox Err.Description & vbCrLf & "Error: " & Err.Number & " (0x" &
Hex(Err.Number) & ")", vbCritical, "Error " & Err.Source
Err.Clear
```

```
End Function
```

```
Public Sub ExecuteMSMQ(ByVal strQueuePath As String, DataToQueue As String)
```

```
Dim QueueMsg As New MSMQMessage
Dim MSMQTx
Dim strData As String
Dim fSend As Boolean
Dim txt As TextStream
Dim mybyte() As Byte
```

```
On Error GoTo Err_SendMSMQ
g_MSMQInfo.FormatName = strQueuePath
Set g_MSMQQueue = g_MSMQInfo.Open(MQ_SEND_ACCESS, MQ_DENY_NONE)
mybyte = StrConv(DataToQueue, vbFromUnicode)
QueueMsg.Body = DataToQueue
```

```
Set MSMQTx = g_MSMTxDisp.BeginTransaction
QueueMsg.Send g_MSMQQueue, MSMQTx
MSMQTx.Commit
```

```
Set QueueMsg = Nothing
Set MSMQTx = Nothing
Exit Sub
```

```
Err_SendMSMQ:
```

```
Err.Raise Err.Number, "Kuyruga mesaji gonderirken", "MSMQ'ya mesaj gonderilemiyor" &
strQueuePath & "" & vbCrLf & vbCrLf & Err.Description
```

```
Exit Sub
```

```
End Sub
```

```
Public Sub PurgeMSMQ(ByVal strQueuePath As String)
```

```
Dim l_MSMQMsg As MSMQMessage
```

```
On Error GoTo Err_ConnectMSMQ
g_MSMQInfo.FormatName = strQueuePath
```

```
Set g_MSMQQueue = g_MSMQInfo.Open(MQ_RECEIVE_ACCESS,
MQ_DENY_NONE)
```

```
On Error GoTo Err_PurgeMSMQ
Do
    Set I_MSMQMsg = g_MSMQQueue.Receive(, , 1)
Loop While Not I_MSMQMsg Is Nothing
```

```
Exit Sub
```

```
Err_ConnectMSMQ:
```

```
Err.Raise Err.Number, "Connecting To MSMQ", "Could Not Open the MSMQ Queue """"
& strQueuePath & """"." & vbCrLf & vbCrLf & Err.Description
```

```
Exit Sub
```

```
Err_PurgeMSMQ:
```

```
Err.Raise Err.Number, "Cleaning MSMQ", "Could Not Remove Existing Messages from
MSMQ Queue """" & strQueuePath & """"." & vbCrLf & vbCrLf & Err.Description
```

```
Exit Sub
```

```
End Sub
```

```
Public Sub ExecuteSchedule(ByVal strSchedule)
```

```
Dim SendPAQ As Object
```

```
On Error GoTo Err_ExecSched
```

```
Set SendPAQ = GetObject(strSchedule)
```

```
If SendPAQ Is Nothing Then
```

```
Err.Raise vbObjectError + 1, , "Invalid Schedule Handle Returned."
```

```
End If
```

```
Set SendPAQ = Nothing
```

```
Exit Sub
```

```
Err_ExecSched:
```

```
Err.Raise Err.Number, "Starting Schedule", "Could Not Launch the XLANG Schedule" &
vbCrLf & "Please verify the path to the SKX file and path to your data are correct. Also
make sure the private queues have been created." & vbCrLf & vbCrLf & Err.Description
```

```
Exit Sub
```

```
End Sub
```

**S1TeklifYazAIC.vbp**

Implements IBTSAppIntegration

Option Explicit

Private rs, con, cmd, ConnectionString

Private m\_DOMDocument As MSXML2.DOMDocument

Private m\_objSubmit As BTSInterchangeLib.Interchange

\*\*\*\*\*

\*\*\*\*\* Gelen Dokuman: T\_AIEm.xml, BizTalk Serverdan alinir.

\*\*\*\*\*

Private Function IBTSAppIntegration\_ProcessMessage( \_  
ByVal bstrDocument As String) As String

Dim j, i, toplam, teklifNo, sqlCode

Dim alEmNo, calisanNo, alEmTarih, kacKalemMal, firma, calisan

Dim kod, satirNo, adet, fiyat, hesaplanmisFiyat

Dim objFileSys, strTimeNow, FileName, fileStream

On Error GoTo hata

ConnectionString = "dsn=Satici1;server=STUDY;database=Satici1;uid=Satici1User;pwd="

Set con = CreateObject("adodb.connection")

Set rs = CreateObject("adodb.recordset")

con.open ConnectionString

rs.ActiveConnection = con

rs.CursorLocation = adUseClient

Set cmd = CreateObject("adodb.command")

cmd.ActiveConnection = ConnectionString

Set m\_DOMDocument = New MSXML2.DOMDocument

m\_DOMDocument.async = False

Set m\_objSubmit = New BTSInterchangeLib.Interchange

IBTSAppIntegration\_ProcessMessage = ""

m\_DOMDocument.loadXML bstrDocument

If m\_DOMDocument.parseError.errorCode <> 0 Then 'The data was not valid XML

Err.Raise m\_DOMDocument.parseError.errorCode, "Satici1 FiyatVer component", \_

"Gelen dokuman islenemiyor : " + \_

vbCrLf + vbCrLf + m\_DOMDocument.parseError.reason + vbCrLf + " satir no: " + \_

CStr(m\_DOMDocument.parseError.Line) + vbCrLf + "yeri: " + \_

CStr(m\_DOMDocument.parseError.linepos)

End If

con.BeginTrans

rs.open ("select \* from T\_teklifNumara"), con, 1

teklifNo = Trim(rs.teklifNo)

teklifNo = teklifNo + 1

con.Execute ("update T\_teklifNumara set teklifNo=" & teklifNo & """)

rs.Close

j = m\_DOMDocument.documentElement.childNodes.length 'node sayisi

```

toplam = 0
i = 2 'mal itemlerinin basladigi node
While i < (j - 1)
    kod = m_DOMDocument.documentElement.childNodes.Item(i).Attributes.Item(1).Text
    rs.open ("select * from T_mal where UniversalKod=" & kod & """), con, 1
    adet = m_DOMDocument.documentElement.childNodes.Item(i).Attributes.Item(3).Text
    fiyat = Trim(rs!brfiyat)
    m_DOMDocument.documentElement.childNodes.Item(i).Attributes.Item(4).Text =
Trim(rs!brfiyat)
    hesaplanmisFiyat = adet * fiyat
    m_DOMDocument.documentElement.childNodes.Item(i).Attributes.Item(5).Text =
hesaplanmisFiyat
    toplam = toplam + hesaplanmisFiyat
    satirNo =
m_DOMDocument.documentElement.childNodes.Item(i).Attributes.Item(6).Text

    sqlCode = "insert into t_teklifAyrinti (teklifNo, universalKod, satirNo, "
    sqlCode = sqlCode & "adet, fiyat, hesaplanmisFiyat )"
    sqlCode = sqlCode & "values (" & teklifNo & ", " & kod & ", "
    sqlCode = sqlCode & satirNo & ", " & adet & ", "
    sqlCode = sqlCode & fiyat & ", " & hesaplanmisFiyat & ")"
    con.Execute sqlCode
    rs.Close
    i = i + 1
Wend
rs.open ("select * from t_teklif where 1=2")
alEmTarih =
m_DOMDocument.documentElement.childNodes.Item(0).Attributes.Item(0).Text
alEmNo =
m_DOMDocument.documentElement.childNodes.Item(0).Attributes.Item(1).Text
firma = m_DOMDocument.documentElement.childNodes.Item(0).Attributes.Item(2).Text
calisan = m_DOMDocument.documentElement.childNodes.Item(1).Attributes.Item(1).Text
& " " & _
    m_DOMDocument.documentElement.childNodes.Item(1).Attributes.Item(0).Text
calisanNo =
m_DOMDocument.documentElement.childNodes.Item(1).Attributes.Item(2).Text
kacKalemMal = m_DOMDocument.documentElement.childNodes.Item(j -
1).Attributes.Item(0).Text
sqlCode = "insert into t_teklif (teklifNo, alEmNo, calisanNo, calisan, "
sqlCode = sqlCode & "alEmTarih, kacKalemMal, alEmToplami, teklifIsteyenFirma )"
sqlCode = sqlCode & "values (" & teklifNo & ", " & alEmNo & ", "
sqlCode = sqlCode & calisanNo & ", " & calisan & ", " & alEmTarih & ", "
sqlCode = sqlCode & kacKalemMal & ", " & toplam & ", "
sqlCode = sqlCode & firma & ")"
con.Execute sqlCode
rs.Close
con.CommitTrans
con.Close

'toplami AEm e ata

```

```

m_DOMDocument.documentElement.childNodes.Item(j - 1).Attributes.Item(1).Text =
toplama
'Satici1 olarak KurumNoyu duzenle.
m_DOMDocument.documentElement.childNodes.Item(0).Attributes.Item(2).Text = "3"
bstrDocument = m_DOMDocument.xml

FileName = "d:\inetpub\wwwroot\tez\Satici1\logSatici1.txt"
strTimeNow = CStr(FormatDateTime(Now(), 1)) & " / " & CStr(FormatDateTime(Now(),
3))
Set objFileSys = CreateObject("Scripting.FileSystemObject")
Set fileStream = objFileSys.OpenTextFile(FileName, ForAppending)
fileStream.WriteLine strTimeNow & " ==> Satici1 teklif gönderdi."
fileStream.Close

Set objFileSys = Nothing
Set fileStream = Nothing

If (firma = "1") Then
  IBTSAppIntegration_ProcessMessage =
m_objSubmit.Submit(BIZTALK_OPENNESS_TYPE_NOTOPEN, bstrDocument, , ,
"T_Satici1 on Seller", , "T_HKK on Seller(Satici1)")
Else
  IBTSAppIntegration_ProcessMessage =
m_objSubmit.Submit(BIZTALK_OPENNESS_TYPE_NOTOPEN, bstrDocument, , ,
"T_Satici1 on Seller", , "T_KKK on Seller(Satici1)")
End If

Set con = Nothing
Set rs = Nothing
Set cmd = Nothing
Set m_DOMDocument = Nothing
Set m_objSubmit = Nothing
Exit Function

hata:
MsgBox Err.Description & vbCrLf & "Error: " & Err.Number & " (0x" &
Hex(Err.Number) & ")", vbCritical, "Error " & Err.Source
Err.Clear

End Function

```

**S1TklfHKKQ.vbp**

Implements IBTSAppIntegration

Option Explicit

Private g\_MSMTxDisp As MSMQ.MSMQTransactionDispenser

Private g\_MSMQQueue As MSMQ.MSMQQueue

Private g\_MSMQInfo As MSMQ.MSMQQueueInfo

Private m\_DOMDocument As MSXML2.DOMDocument

Private g\_SkedFile As String

\*\*\*\*\*

\*\*\*\*\* Gelen Dokuman: T\_alEm.xml, BizTalk Serverdan alinir.

\*\*\*\*\*

Private Function IBTSAppIntegration\_ProcessMessage(  
ByVal bstrDocument As String) As String

Dim objFileSys, strTimeNow, FileName, fileStream

On Error GoTo hata

Set m\_DOMDocument = New MSXML2.DOMDocument

m\_DOMDocument.async = False

IBTSAppIntegration\_ProcessMessage = ""

Set g\_MSMQInfo = CreateObject("MSMQ.MSMQQueueInfo")

Set g\_MSMTxDisp = CreateObject("MSMQ.MSMQTransactionDispenser")

m\_DOMDocument.loadXML bstrDocument

If m\_DOMDocument.parseError.errorCode &lt;&gt; 0 Then "The data was not valid XML

Err.Raise m\_DOMDocument.parseError.errorCode, "S1TeklifHKKQ component", \_

"Gelen dokuman islenemiyor :" + \_

vbCrLf + vbCrLf + m\_DOMDocument.parseError.reason + vbCrLf + " satir no: " + \_

CStr(m\_DOMDocument.parseError.Line) + vbCrLf + "yeri: " + \_

CStr(m\_DOMDocument.parseError.linepos)

End If

'satici1 schedule

g\_SkedFile = "sked:/// " &amp; "d:\inetpub\wwwroot\tez\Satici1\Schedule\S1HKKTeklif.skx"

'Connect To MSMQ and Remove Any Existing Messages

PurgeMSMQ "DIRECT=OS:\private\$\S1Teklif"

'Send Selected message to MSMQ

ExecuteMSMQ "DIRECT=OS:\private\$\S1Teklif", bstrDocument

ExecuteSchedule g\_SkedFile

Set objFileSys = Nothing

FileName = "d:\inetpub\wwwroot\tez\Satici1\logSatici1.txt"

strTimeNow = CStr(FormatDateTime(Now(), 1)) &amp; " / " &amp; CStr(FormatDateTime(Now(),

3))

Set objFileSys = CreateObject("Scripting.FileSystemObject")

Set fileStream = objFileSys.OpenTextFile(FileName, ForAppending)

fileStream.WriteLine strTimeNow &amp; " ==&gt; S1Teklif kuyruguna HKK dokumani atandi."

fileStream.WriteLine strTimeNow &amp; " ==&gt; Satici1 TeklifHKK Schedule aktif edildi."

```
fileStream.Close
```

```
IBTSAppIntegration_ProcessMessage = bstrDocument
```

```
Set objFileSys = Nothing
Set fileStream = Nothing
Set g_MSMQInfo = Nothing
Set g_MSMTxDisp = Nothing
Set m_DOMDocument = Nothing
Exit Function
```

```
hata:
```

```
MsgBox Err.Description & vbCrLf & "Error: " & Err.Number & " (0x" &
Hex(Err.Number) & ")", vbCritical, "Error " & Err.Source
Err.Clear
```

```
End Function
```

```
Public Sub ExecuteMSMQ(ByVal strQueuePath As String, DataToQueue As String)
```

```
Dim QueueMsg As New MSMQMessage
Dim MSMQTx
Dim strData As String
Dim fSend As Boolean
Dim txt As TextStream
Dim mybyte() As Byte
```

```
On Error GoTo Err_SendMSMQ
g_MSMQInfo.FormatName = strQueuePath
Set g_MSMQQueue = g_MSMQInfo.Open(MQ_SEND_ACCESS, MQ_DENY_NONE)
mybyte = StrConv(DataToQueue, vbFromUnicode)
QueueMsg.Body = DataToQueue
```

```
Set MSMQTx = g_MSMTxDisp.BeginTransaction
QueueMsg.Send g_MSMQQueue, MSMQTx
MSMQTx.Commit
```

```
Set QueueMsg = Nothing
Set MSMQTx = Nothing
Exit Sub
```

```
Err_SendMSMQ:
```

```
Err.Raise Err.Number, "Kuyruğa mesajı gönderirken", "MSMQ'ya mesaj gönderilemiyor" &
strQueuePath & """" & vbCrLf & vbCrLf & Err.Description
Exit Sub
End Sub
```

```
Public Sub PurgeMSMQ(ByVal strQueuePath As String)
```

```
Dim l_MSMQMsg As MSMQMessage

On Error GoTo Err_ConnectMSMQ
g_MSMQInfo.FormatName = strQueuePath
```

```
Set g_MSMQQueue = g_MSMQInfo.Open(MQ_RECEIVE_ACCESS,
MQ_DENY_NONE)
```

```
On Error GoTo Err_PurgeMSMQ
```

```
Do
```

```
Set l_MSMQMsg = g_MSMQQueue.Receive(, , 1)
```

```
Loop While Not l_MSMQMsg Is Nothing
```

```
Exit Sub
```

```
Err_ConnectMSMQ:
```

```
Err.Raise Err.Number, "Connecting To MSMQ", "Could Not Open the MSMQ Queue """"
& strQueuePath & """"." & vbCrLf & vbCrLf & Err.Description
```

```
Exit Sub
```

```
Err_PurgeMSMQ:
```

```
Err.Raise Err.Number, "Cleaning MSMQ", "Could Not Remove Existing Messages from
MSMQ Queue """" & strQueuePath & """"." & vbCrLf & vbCrLf & Err.Description
```

```
Exit Sub
```

```
End Sub
```

```
Public Sub ExecuteSchedule(ByVal strSchedule)
```

```
Dim SendPAQ As Object
```

```
On Error GoTo Err_ExecSched
```

```
Set SendPAQ = GetObject(strSchedule)
```

```
If SendPAQ Is Nothing Then
```

```
Err.Raise vbObjectError + 1, , "Invalid Schedule Handle Returned."
```

```
End If
```

```
Set SendPAQ = Nothing
```

```
Exit Sub
```

```
Err_ExecSched:
```

```
Err.Raise Err.Number, "Starting Schedule", "Could Not Launch the XLANG Schedule" &
vbCrLf & "Please verify the path to the SKX file and path to your data are correct. Also
make sure the private queues have been created." & vbCrLf & vbCrLf & Err.Description
```

```
Exit Sub
```

```
End Sub
```



**S1TeklifKKKQ.vbp**

Implements IBTSAppIntegration

Option Explicit

Private g\_MSMTxDisp As MSMQ.MSMQTransactionDispenser

Private g\_MSMQQueue As MSMQ.MSMQQueue

Private g\_MSMQInfo As MSMQ.MSMQQueueInfo

Private m\_DOMDocument As MSXML2.DOMDocument

Private g\_SkedFile As String

\*\*\*\*\*

\*\*\*\*\* Gelen Dokuman: T\_alEm.xml, BizTalk Serverdan alinir.

\*\*\*\*\*

Private Function IBTSAppIntegration\_ProcessMessage(\_  
ByVal bstrDocument As String) As String

Dim objFileSys, strTimeNow, FileName, fileStream

On Error GoTo hata

Set m\_DOMDocument = New MSXML2.DOMDocument

m\_DOMDocument.async = False

IBTSAppIntegration\_ProcessMessage = ""

Set g\_MSMQInfo = CreateObject("MSMQ.MSMQQueueInfo")

Set g\_MSMTxDisp = CreateObject("MSMQ.MSMQTransactionDispenser")

m\_DOMDocument.loadXML bstrDocument

If m\_DOMDocument.parseError.errorCode <> 0 Then "The data was not valid XML

Err.Raise m\_DOMDocument.parseError.errorCode, "S1TeklifKKKQ component", \_

"Gelen dokuman islenemiyor ." + \_

vbCrLf + vbCrLf + m\_DOMDocument.parseError.reason + vbCrLf + " satir no: " + \_

CStr(m\_DOMDocument.parseError.Line) + vbCrLf + "yeri: " + \_

CStr(m\_DOMDocument.parseError.linepos)

End If

'satici1 schedule

g\_SkedFile = "sked:/// " & "d:\inetpub\wwwroot\tez\Satici1\Schedule\S1KKKTeklif.skx"

'Connect To MSMQ and Remove Any Existing Messages

PurgeMSMQ "DIRECT=OS:\private\$\S1Teklif"

'Send Selected message to MSMQ

ExecuteMSMQ "DIRECT=OS:\private\$\S1Teklif", bstrDocument

ExecuteSchedule g\_SkedFile

Set objFileSys = Nothing

FileName = "d:\inetpub\wwwroot\tez\Satici1\logSatici1.txt"

strTimeNow = CStr(FormatDateTime(Now(), 1)) & " / " & CStr(FormatDateTime(Now(),

3))

Set objFileSys = CreateObject("Scripting.FileSystemObject")

Set fileStream = objFileSys.OpenTextFile(FileName, ForAppending)

fileStream.WriteLine strTimeNow & " ==> S1Teklif kuyruguna KKK dokumani atandi."

fileStream.WriteLine strTimeNow & " ==> Satici1 TeklifKKK Schedule aktif edildi."

```
fileStream.Close
```

```
IBTSAppIntegration_ProcessMessage = bstrDocument
```

```
Set objFileSys = Nothing
Set fileStream = Nothing
Set g_MSMQInfo = Nothing
Set g_MSMTxDisp = Nothing
Set m_DOMDocument = Nothing
Exit Function
```

```
hata:
```

```
MsgBox Err.Description & vbCrLf & "Error: " & Err.Number & " (0x" &
Hex(Err.Number) & ")", vbCritical, "Error " & Err.Source
Err.Clear
```

```
End Function
```

```
Public Sub ExecuteMSMQ(ByVal strQueuePath As String, DataToQueue As String)
```

```
Dim QueueMsg As New MSMQMessage
Dim MSMQTx
Dim strData As String
Dim fSend As Boolean
Dim txt As TextStream
Dim mybyte() As Byte
```

```
On Error GoTo Err_SendMSMQ
g_MSMQInfo.FormatName = strQueuePath
Set g_MSMQQueue = g_MSMQInfo.Open(MQ_SEND_ACCESS, MQ_DENY_NONE)
mybyte = StrConv(DataToQueue, vbFromUnicode)
QueueMsg.Body = DataToQueue
```

```
Set MSMQTx = g_MSMTxDisp.BeginTransaction
QueueMsg.Send g_MSMQQueue, MSMQTx
MSMQTx.Commit
```

```
Set QueueMsg = Nothing
Set MSMQTx = Nothing
Exit Sub
```

```
Err_SendMSMQ:
```

```
Err.Raise Err.Number, "Kuyruga mesaji gonderirken", "MSMQ'ya mesaj gonderilemiyor" &
strQueuePath & """" & vbCrLf & vbCrLf & Err.Description
Exit Sub
End Sub
```

```
Public Sub PurgeMSMQ(ByVal strQueuePath As String)
```

```
Dim l_MSMQMsg As MSMQMessage

On Error GoTo Err_ConnectMSMQ
g_MSMQInfo.FormatName = strQueuePath
```

```
Set g_MSMQQueue = g_MSMQInfo.Open(MQ_RECEIVE_ACCESS,
MQ_DENY_NONE)
```

```
On Error GoTo Err_PurgeMSMQ
Do
    Set l_MSMQMsg = g_MSMQQueue.Receive(, , 1)
Loop While Not l_MSMQMsg Is Nothing
```

```
Exit Sub
```

```
Err_ConnectMSMQ:
```

```
Err.Raise Err.Number, "Connecting To MSMQ", "Could Not Open the MSMQ Queue """"
& strQueuePath & """"." & vbCrLf & vbCrLf & Err.Description
```

```
Exit Sub
```

```
Err_PurgeMSMQ:
```

```
Err.Raise Err.Number, "Cleaning MSMQ", "Could Not Remove Existing Messages from
MSMQ Queue """" & strQueuePath & """"." & vbCrLf & vbCrLf & Err.Description
```

```
Exit Sub
```

```
End Sub
```

```
Public Sub ExecuteSchedule(ByVal strSchedule)
```

```
Dim SendPAQ As Object
```

```
On Error GoTo Err_ExecSched
```

```
Set SendPAQ = GetObject(strSchedule)
```

```
If SendPAQ Is Nothing Then
```

```
Err.Raise vbObjectError + 1, , "Invalid Schedule Handle Returned."
```

```
End If
```

```
Set SendPAQ = Nothing
```

```
Exit Sub
```

```
Err_ExecSched:
```

```
Err.Raise Err.Number, "Starting Schedule", "Could Not Launch the XLANG Schedule" &
vbCrLf & "Please verify the path to the SKX file and path to your data are correct. Also
make sure the private queues have been created." & vbCrLf & vbCrLf & Err.Description
```

```
Exit Sub
```

```
End Sub
```

## Ek 8 Windows Script Components

### *HKKAIEmIstYaz.wsc*

```

<?xml version="1.0"?>
<component>
<?component error="true" debug="true"?>
<registration
  description="HKKAIEmIstKaydet"
  progid="Tezim.HKKAIEmIstKaydet"
  version="1.00"
  classid="{2938621b-40fc-48e2-827c-bed74e21a540}"
>
</registration>
<public>
  <method name="yazAIEmIst">
    <PARAMETER name="Document"/>
    <PARAMETER name="FileName"/>
  </method>
</public>
<implements type="Behavior" id="Behavior"/>
<script language="VBScript">
<![CDATA[

function yazAIEmIst(Document, FileName)
  dim objFileSys, strTimeNow
  if FileName = "" then
    FileName = "d:\inetpub\wwwroot\tez\AIEmIst\alEm*.xml"
  end if

  strTimeNow = left(Replace(CStr(FormatDateTime(Now(), 3)), ":", ""), 6)
  Set objFileSys = CreateObject("Scripting.FileSystemObject")
  Set filestream = objFileSys.CreateTextFile(Replace(FileName, "*", strTimeNow), False)
  filestream.writeline Document
  filestream.close

  set objFileSys = Nothing

end function
]]>
</script>
</component>

```

***HKKTeklifYaz.wsc***

```

<?xml version="1.0"?>
<component>
<?component error="true" debug="true"?>
<registration
  description="HKKTeklifKaydet"
  progid="Tezim.HKKTeklifKaydet"
  version="1.00"
  classid="{2938621b-40fc-48e2-827c-bed74e21a541}"
>
</registration>
<public>
  <method name="yazTeklif">
    <PARAMETER name="S1Document"/>
    <PARAMETER name="S2Document"/>
    <PARAMETER name="S3Document"/>

  </method>
</public>
<implements type="Behavior" id="Behavior"/>
<script language="VBScript">
<![CDATA[

function yazTeklif(S1Document, S2Document, S3Document)
  dim objFileSys, strTimeNow, FileName, teklifYaz

  strTimeNow = left(Replace(CStr(FormatDateTime(Now(), 3)), ":", ""), 6)

  FileName = "d:\inetpub\wwwroot\tez\teklifS1\tklf*.xml"
  Set objFileSys = CreateObject("Scripting.FileSystemObject")
  Set filestream = objFileSys.CreateTextFile(Replace(FileName, "*", strTimeNow), False)
  filestream.writeline S1Document
  filestream.close
  Set objFileSys = Nothing

  FileName = "d:\inetpub\wwwroot\tez\teklifS2\tklf*.xml"
  Set objFileSys = CreateObject("Scripting.FileSystemObject")
  Set filestream = objFileSys.CreateTextFile(Replace(FileName, "*", strTimeNow), False)
  filestream.writeline S2Document
  filestream.close
  Set objFileSys = Nothing

  FileName = "d:\inetpub\wwwroot\tez\teklifS3\tklf*.xml"
  Set objFileSys = CreateObject("Scripting.FileSystemObject")
  Set filestream = objFileSys.CreateTextFile(Replace(FileName, "*", strTimeNow), False)
  filestream.writeline S3Document
  filestream.close

  Set teklifYaz = CreateObject("HKKTeklif.TeklifKaydet")
  teklifYaz.s1TeklifKaydet(S1Document)

```

```
teklifYaz.s2TeklifKaydet(S2Document)
teklifYaz.s3TeklifKaydet(S3Document)
MsgBox "Saticilarin verdigi teklifler veritabanina kaydedildi. TeklifS1, TeklifS2, TeklifS3
dizinlerine de XML olarak kaydedildi." + vbcrLf + + vbcrLf , 0, "Teklif Yazma islemi bitti."
```

```
set objFileSys = Nothing
set teklifYaz = Nothing
```

```
end function
]]>
</script>
</component>
```



**Ek 9 Organizasyon Tanımları**

T\_HKK on Buyer  
T\_HKK on Seller(Satici1)  
T\_HKK on Seller(Satici2)  
T\_HKK on Seller(Satici3)  
T\_KKK on Buyer  
T\_KKK on Seller(Satici1)  
T\_KKK on Seller(Satici2)  
T\_KKK on Seller(Satici3)  
T\_Satici1 on Buyer(HKK)  
T\_Satici1 on Buyer(KKK)  
T\_Satici1 on Seller  
T\_Satici2 on Buyer(HKK)  
T\_Satici2 on Buyer(KKK)  
T\_Satici2 on Seller  
T\_Satici3 on Buyer(HKK)  
T\_Satici3 on Buyer(KKK)  
T\_Satici3 on Seller



## Ek 10 Dosya Alım Fonksiyonu Tanımları

Adı	Kullandığı Kanal İsmi	Dinlediği Dizin
KKK Tklflste	K_KKKTklflste	d:\inetpub\wwwroot\tez\KKK\AIEm\AIEm*.xml
S1 KKKSprsVer	K_S1KKKSprVer	d:\inetpub\wwwroot\tez\satici1\siparis\KKKSpr*.xml
S1 KKKTklfVer	K_S1KKKTklfVer	d:\inetpub\wwwroot\tez\satici1\teklif\KKKTklf*.xml
KKK S1Sprs	K_S1SprsBilgEkle	d:\inetpub\wwwroot\tez\KKK\SiparisS1\ilkSprs*.xml
S2 KKKSprsVer	K_S2KKKSprVer	d:\inetpub\wwwroot\tez\satici2\siparis\KKKSpr*.xml
S2 KKKTklfVer	K_S2KKKTklfVer	d:\inetpub\wwwroot\tez\satici2\teklif\KKKTklf*.xml
KKK S2Sprs	K_S2SprsBilgEkle	d:\inetpub\wwwroot\tez\KKK\SiparisS2\ilkSprs*.xml
S3 KKKSprsVer	K_S3KKKSprVer	d:\inetpub\wwwroot\tez\satici3\siparis\KKKSpr*.xml
S3 KKKTklfVer	K_S3KKKTklfVer	d:\inetpub\wwwroot\tez\satici3\teklif\KKKTklf*.xml
KKK S3Sprs	K_S3SprsBilgEkle	d:\inetpub\wwwroot\tez\KKK\SiparisS3\ilkSprs*.xml
HKK Tklflste	T_HKKTklflste	d:\inetpub\wwwroot\tez\AIEm\AIEm*.xml
S1 HKKSprVer	T_S1HKKSprVer	d:\inetpub\wwwroot\tez\Satici1\Siparis\Spr*.xml
S1 HKKTklfVer	T_S1HKKTklfVer	d:\inetpub\wwwroot\tez\Satici1\Teklif\Tklf*.xml
HKK S1Sprs	T_S1SprsBilgEkle	d:\inetpub\wwwroot\tez\SiparisS1\ilkSprs*.xml
S2 HKKSprVer	T_S2HKKSprVer	d:\inetpub\wwwroot\tez\Satici2\Siparis\Spr*.xml
S2 HKKTklfVer	T_S2HKKTklfVer	d:\inetpub\wwwroot\tez\Satici2\Teklif\Tklf*.xml
HKK S2Sprs	T_S2SprsBilgEkle	d:\inetpub\wwwroot\tez\SiparisS2\ilkSprs*.xml
S3 HKKSprVer	T_S3HKKSprVer	d:\inetpub\wwwroot\tez\Satici3\Siparis\Spr*.xml
S3 HKKTklfVer	T_S3HKKTklfVer	d:\inetpub\wwwroot\tez\Satici3\Teklif\Tklf*.xml
HKK S3Sprs	T_S3SprsBilgEkle	d:\inetpub\wwwroot\tez\SiparisS3\ilkSprs*.xml



## Ek 11 Port Tanımları

Adı	Temel Gönderim
K_Port AIEmlstToAlem	file:///d:/inetpub/wwwroot/tez\KKK\AIEmlalEm%tracking_id%.xml
K_Port KKKTkflste	KKKTkflste Sprs
K_Port S1AIEmlToFat	file:///d:/inetpub/wwwroot/tez\KKK\SiparisS1\ilkSprs%tracking_id%.xml
K_Port S1Fat	file:///d:/inetpub/wwwroot/tez\KKK\fatura\fat%tracking_id%.xml
K_Port S1KKKSprVer	S1SprsKKKQ SprsSchedule
K_Port S1KKKTkflVer	S1TeklifKKKQ TkflSchedule
K_Port S1SprsBilgEkle	KKKSprS1 SprsS1
K_Port S1SprsV	S1SprsKKKAIC Satici1Sprs
K_Port S1Teklif	http://study/KS1FiyatQ/satici1MSMQ.asp
K_Port S1Tkfl	S1TeklifYazAIC S1Teklif
K_Port S2AIEmlToFat	file:///d:/inetpub/wwwroot/tez\KKK\SiparisS2\ilkSprs%tracking_id%.xml
K_Port S2Fat	file:///d:/inetpub/wwwroot/tez\KKK\fatura\fat%tracking_id%.xml
K_Port S2KKKSprVer	S2SprsKKKQ SprsSchedule
K_Port S2KKKTkflVer	S2TeklifKKKQ TkflSchedule
K_Port S2SprsBilgEkle	KKKSprS2 SprsS2
K_Port S2SprsV	S2SprsKKKAIC Satici2Sprs
K_Port S2Teklif	http://study/KS2FiyatQ/satici2MSMQ.asp
K_Port S2Tkfl	S2TeklifYazAIC S2Teklif
K_Port S3AIEmlToFat	file:///d:/inetpub/wwwroot/tez\KKK\SiparisS3\ilkSprs%tracking_id%.xml
K_Port S3Fat	file:///d:/inetpub/wwwroot/tez\KKK\fatura\fat%tracking_id%.xml
K_Port S3KKKSprVer	S3SprsKKKQ SprsSchedule
K_Port S3KKKTkflVer	S3TeklifKKKQ TkflSchedule
K_Port S3SprsBilgEkle	KKKSprS3 SprsS3
K_Port S3SprsV	S3SprsKKKAIC Satici3Sprs
K_Port S3Teklif	http://study/KS3FiyatQ/satici3MSMQ.asp
K_Port S3Tkfl	S3TeklifYazAIC S3Teklif
T_Port AIEmlstToAIEml	file:///d:/inetpub/wwwroot/tez\AIEmlalEm%tracking_id%.xml
T_Port HKKTkflste	HKKTkflste Sprs
T_Port S1AIEmlToFat	file:///d:/inetpub/wwwroot/tez\SiparisS1\ilkSprs%tracking_id%.xml
T_Port S1Fat	file:///d:/inetpub/wwwroot/tez\fatura\fat%tracking_id%.xml
T_Port S1HKKSprVer	S1SprsHKKQ SprsSchedule
T_Port S1HKKTkflVer	S1TeklifHKKQ TkflSchedule
T_Port S1SprsBilgEkle	HKKSprS1 SprsS1
T_Port S1SprsV	S1SprsHKKAIC Satici1Sprs
T_Port S1Teklif	http://study/HS1FiyatQ/satici1MSMQ.asp
T_Port S1Tkfl	S1TeklifYazAIC S1Teklif
T_Port S2AIEmlToFat	file:///d:/inetpub/wwwroot/tez\SiparisS2\ilkSprs%tracking_id%.xml
T_Port S2Fat	file:///d:/inetpub/wwwroot/tez\fatura\fat%tracking_id%.xml
T_Port S2HKKSprVer	S2SprsHKKQ SprsSchedule
T_Port S2HKKTkflVer	S2TeklifHKKQ TkflSchedule
T_Port S2SprsBilgEkle	HKKSprS2 SprsS2
T_Port S2SprsV	S2SprsHKKAIC Satici2Sprs
T_Port S2Teklif	http://study/HS2FiyatQ/satici2MSMQ.asp
T_Port S2Tkfl	S2TeklifYazAIC S2Teklif

<b>Adı</b>	<b>Temel Gönderim</b>
T_Port S3AlEmToFat	file://d:\inetpub\wwwroot\tez\SiparisS3\ilkSprs%tracking_id%.xml
T_Port S3Fat	file://d:\inetpub\wwwroot\tez\fatura\fat%tracking_id%.xml
T_Port S3HKKSprVer	S3SprsHKKQ SprsSchedule
T_Port S3HKKTklfVer	S3TeklifHKKQ TklfSchedule
T_Port S3SprsBilgEkle	HKKSprS3 SprsS3
T_Port S3SprsV	S3SprsHKKAIC Satici3Sprs
T_Port S3Teklif	<a href="http://study/HS3FiyatQ/satici3MSMQ.asp">http://study/HS3FiyatQ/satici3MSMQ.asp</a>
T_Port S3Tklf	S3TeklifYazAIC S3Teklif

**Ek 12 Kuyruk Tanımları**

HalEmIst	Alım Emri İsteđi için
HKKS1	Satıcı1 kurumundan HKK'ya gelen teklif için
HKKS2	Satıcı2 kurumundan HKK'ya gelen teklif için
HKKS3	Satıcı3 kurumundan HKK'ya gelen teklif için
KA1EmIst	Alım Emri İsteđi için
KKKS1	Satıcı1 kurumundan KKK'ya gelen teklif için
KKKS2	Satıcı2 kurumundan KKK'ya gelen teklif için
KKKS3	Satıcı3 kurumundan KKK'ya gelen teklif için
S1Siparis	Satıcı1 kurumuna verilen siparişler için
S1Teklif	Satıcı1 kurumuna verilen teklifler için
S2Siparis	Satıcı2 kurumuna verilen siparişler için
S2Teklif	Satıcı2 kurumuna verilen teklifler için
S3Siparis	Satıcı3 kurumuna verilen siparişler için
S3Teklif	Satıcı3 kurumuna verilen teklifler için

**Ek 13 ODBC Tanımları**

HKK  
KKK  
Satici1  
Satici2  
Satici3



**Ek 14 Virtual Directory Tanımları**

HS1FiyatQ	D:\Inetpub\wwwroot\tez\V_S1FiyatQ
HS2FiyatQ	D:\Inetpub\wwwroot\tez\V_S2FiyatQ
HS3FiyatQ	D:\Inetpub\wwwroot\tez\V_S3FiyatQ
KS1FiyatQ	D:\Inetpub\wwwroot\tez\KKK\V_KS1FiyatQ
KS2FiyatQ	D:\Inetpub\wwwroot\tez\KKK\V_KS2FiyatQ
KS3FiyatQ	D:\Inetpub\wwwroot\tez\KKK\V_KS3FiyatQ
Tez	D:\Inetpub\wwwroot\tez
KKK	D:\Inetpub\wwwroot\tez\KKK
Satici1	D:\Inetpub\wwwroot\tez\Satici1
Satici2	D:\Inetpub\wwwroot\tez\Satici2
Satici3	D:\Inetpub\wwwroot\tez\Satici3



## Ek 15 Elektronik Ticaret Terimler Sözlüğü (Bozkurt, 2000)

**açık anahtar (public key):** Açık anahtarlı bir şifreleyici yöntem (algoritma) kullanan bir kullanıcının kendisine ait olan iki anahtarından, kamuya açık olanı.

**açık anahtar altyapısı-AAA (public key infrastructure-PKI):** Bilgi iletişimde açık anahtarlı kriptografinin yaygın ve güvenli olarak kullanılabilmesini sağlamaya yarayan ve birbirleriyle eşgüdüm içinde çalışan anahtar üretimi, anahtar yönetimi, onay kurumu, sayısal noterlik, zaman damgası gibi hizmetlerin tümü.

**açık anahtarlı kriptografi (public key cryptography):** Her kullanıcıya, sürekli kullanım için biri açık diğeri gizli iki anahtarın verildiği şifreleme/şifre çözme yöntemlerinin tümü. Asimetrik şifreleme ya da çift anahtarlı şifreleme adını da almaktadır.

**açık bilgisayar ağı (open computer network):** İsteyen herhangi bir bilgisayar kullanıcısının bağlanabileceği ve diğer kişilerle bilgisayar üzerinden iletişim kurabileceği, herkese açık elektronik iletişim ortamı.

**anahtar (key):** Şifreleme ve şifre çözme sırasında kullanılan sayı dizisi.

**anahtar üretimi (key generation):** Açık anahtarlı şifrelemede, her kullanıcının açık/gizli anahtar çiftinin, kullanılan şifreleyici yönteme bağlı matematiksel işlemlerle hazırlanması.

**anahtar yönetimi (key management):** Açık anahtarlı şifrelemede her kullanıcıya farklı anahtar çiftleri verilmesi, kullanıcıların açık anahtarlarının herkesin ulaşımına açık olarak saklanması ve kullanıcıların gizli anahtarlarının mutlak gizliliğinin sağlanmasından sorumlu düzen.

**anahtarı bulan kurum-ABK (key recovery agency-KRA):** Yasal erişime yardımcı olmak amacıyla kurulan ve yargının gerektirdiği durumlarda, zan altındaki kişinin gizli anahtarının matematiksel yöntemlerle elde edilmesini sağlayan kurum. Gizli anahtarını kaybeden herhangi bir kişi de, kimliğini belgeleyerek ABK'ya başvurursa anahtarını yeniden elde edebilir.

**basit bölüm (Simple segment):** Hiçbir sınıflandırmaya ihtiyacı olmayan bölüm (Anlamı sabit ve açık olan bölüm).

**basit veri elemanı (Simple data element):** Tek bir değer taşıyan veri elemanı.

**bilgi bütünlüğü (message integrity):** Bilginin saklanması veya açık/kapalı iletişim ağlarından iletimi sırasında içerik açısından herhangi bir değişime uğratılmamış olması, özgün halinde korunması.

**bilgi güvenliği (information security):** Bilginin, i) kime ait olduğu belirlenmiş, ii) bütünlüğü korunarak, iii) gizliliği sağlanmış olarak iletimi ve saklanması.

**bölüm adı (Segment name):** Doğal dilde bir ya da daha çok sözcük ile veri bölümü kavramının tanımlanması.

**bölüm kodu (Segment code):** Bölüm rehberinde tanımlandığı şekilde, her bölümü tek olarak gösteren kod.

**bölüm rehberi (Segment directory):** Tanımlandırılmış, isimlendirilmiş bölümler listesi.

**çift anahtarlı kriptografi (double key cryptography):** Açık anahtarlı şifreleme veya asimetric şifreleme.

**doküman (Document):** Bir verinin üzerine kayıt edildiği, insan ya da makine tarafından okunabilen, (değişmez) veri taşıyıcı.

**eCo Document Wrapper:** Doküman hakkında basit meta veri kümesi.

**elektronik kimlik belgesi-EKB (digital certificate):** Onay kurumunun hazırladığı ve sayısal olarak imzaladığı, hangi açık anahtarın hangi kişiye ait olduğunu gösteren belge.

**elektronik veri değişimi-EVD (electronic data interchange-EDI):** Standart bir formda yazılmış olan bilgilerin bilgisayarlar arasında aktarımı ve otomatik olarak yorumlanıp işlenebilmesi.

**Elektronik Veri Değişimi (Electronic Data Interchange):** Standart bir yapıda bilgisayardan – bilgisayara veri (ticari) transferi.

**erişim (access):** Herhangi bir sistemi kullanmaya başlama, örneğin bir elektronik ticaret sistemine bilgisayar üzerinden bağlanarak iletişim kurma.

**EVD kurumu (EDI association):** Bir ülkede EVD kullanımını düzenleyen kurum, örneğin, ABD'deki EDIA, Avustralya'daki EDICA, Kanada'daki EDICC veya Yeni Zelanda'daki EDIANZ.

**EVD servis sunucusu (EDI server):** Bir EVD servisinin merkezinde olan bilgisayar sistemi.

**gizli-özel, kişisel-anahtar (private key):** Açık anahtarlı şifreleme kullanan bir kullanıcının, kendisine ait olan iki anahtarından gizli tutulanı.

**gizlilik (privacy):** İletişim kuran iki taraf arasındaki yazışmaların, üçüncü kişilerden gizli tutulması, veya bir kişiye ait bilgilerin kendisi dışında herkesten gizli tutulması.

**güvenilir üçüncü kuruluş, kurum veya kişi-GÜK (trusted third party-TTP):** Bir çeşit onay kurumu. Onay kurumlarının yaptığı gibi kişilerin kimliğini güvenli olarak belirleyip, elektronik kimlik belgelerini hazırlamaya ve anahtar yönetimini sağlamaya ek olarak, kişilerin gizli anahtarlarını çok güvenli bir ortamda saklayan ve gerektiğinde yargı kararıyla yetkili makamlara veren kuruluş.

**kanal (channel):** Bilginin bir kullanıcıdan diğerine iletimi için gereken fiziksel iletişim ortamı, örneğin, bilgisayar bağlantısı, telefon kablosu, radyolink ve uydu üzerinden diğer kullanıcıya ulaşan bağlantının tümü.

**kapalı bilgisayar ağı (closed computer network):** Kullanıcılarından biri olmak için belirli koşulların sağlanması gerektiği, herkese açık olmayan bilgisayar ağları. Örnek: Bankalar ve bankamatikler arasındaki bağlantı.

**kimlik belirleme (authentication):** Herhangi bir servisi almak isteyen birinin, gerçekten de kendi iddia ettiği kişi olduğunun belirlenmesi.

**kod:** (a) Bilginin kısaltılarak kayıt edildiği ya da tanımlandığı karakter dizisi (b) Bilgisayarın tanyacağı formda özel semboller kullanılarak bilginin gösterilmesi ya da tanımlanması.

**kriptografik algoritma (cryptographic algorithm):** Şifreleme/şifre çözmede kullanılan belirli bir yöntemin ayrıntılı içeriği, bu içeriğin matematiksel adımları.

**kriptoloji (cryptology):** Güvenli bilgi iletişimi ve/veya saklanması için şifreleme ve şifre çözme yöntemleri türeten, geliştiren, inceleyen bilim dalı.

**mesaj çizeneği (Message diagram):** Bir mesaj içindeki bölüm dizisinin grafiksel gösterimi.

**mesaj kodu (Message code):** Mesaj tipini tanımlayan ve tek olan alfabetik referans (isim).

**mesaj rehberi (Message directory):** İsimlendirilmiş, tanımlanmış ve tarif edilmiş mesaj tiplerinin listesi.

**mesaj tipi (Message type):** Belirlenmiş işlem tipi için ihtiyaçları kapsayan, tanımlanmış ve planlanmış veri kümesi (seti).

**mesaj (Message):** Bilgiyi taşımak üzere planlanmış sıralı (düzenli) karakter serisi.

**meta veri:** Diğer veriyi tanımlamak için kullanılan veri.

**onay kurumu-OK (certifying authority-CA):** Kişilerin kimliğini güvenli olarak belirleyip elektronik kimlik belgelerini hazırlayan ve anahtar yönetimini sağlayan kuruluş.

**sayısal imza (digital signature):** Elektronik ortamdaki yazışmalara eklenen, yazıyı gönderenin kimliğini ve gönderilen yazının iletim sırasında bozulmadığını kanıtlamaya yarayan bölüm. Sayısal imza, yazının içeriğine ve imzalayanın gizli anahtarına bağlı bir şifreleyici yöntemle atıldığı için, sayısal imzanın doğrulanmasında, imzayı atanın açık anahtarı kullanılır.

**sayısal noter (digital notary):** Bilgisayar ağlarında iletilen bilgileri tarafların isteği ile saklayıp, kendisine başvurulduğunda belgeleyebilen kuruluş.

**şema:** Nesne sınıfının özelliklerini tanımlar.

**tek anahtarlı kriptografi (single key cryptography):** Şifreleme ve şifre çözme için aynı anahtarı kullanan şifreleyici yöntemlerin tümü. Simetrik şifreleme veya gizli anahtarlı şifreleme adını da alır. Kullanılan gizli anahtarı mesajı gönderen ve alan kişilerin paylaşması gerektiği için, tek anahtarlı şifrelemenin güvenilirliği, her kullanıcı çiftine ayrı bir anahtar verilebilmesine bağlıdır. Bu durumda, bir kullanıcı, haberleşeceği herkes için farklı bir anahtar kullanmak zorundadır; bu ise önemli bir anahtar dağıtım problemiye yol açar. Çift anahtarlı şifreleme, bu sorunu ortadan kaldırmıştır.

**UN/EDIFACT :** Mesaj rehberinde belirlendiği düzende sıralı bölümler kümesi.

**veri elemanı adı (Data element name):** Doğal dilde bir ya da daha çok sözcük ile veri elemanı kavramının tanımlanması.

**veri elemanı niteliği (Data element attribute):** Veri elemanının tanımlanmış özelliği.

**veri elemanı rehberi (Data element directory):** Tanımlanmış, isimlendirilmiş veri elemanı niteliklerinin, uygun veri elemanı değerinin nasıl simgelenmesine ilişkin özellikleri içeren liste.

**veri elemanı:** Verinin, tanımlamak, değer göstermek için özellikleri belirlenmiş bir birimi.

**veri:** Bilginin, iletişim, yorum ya da işlem için uygun olarak formüllemiş şekilde gösterilmesi.



**yasal erişim (lawful access):** Devletin, açık anahtarlı bir şifreleyici algoritma kullananların gizli anahtarlarına, yasaların gerektirdiği durumlarda ve yargı kararıyla ulaşabilme yetkisi.

**zaman damgası (time stamp):** Bilgisayar ağlarında iletilen mesajlara eklenen ve mesajın yazıldığı zamanı güvenli olarak belgeleyen damga.



**ÖZGEÇMİŞ**

Doğum tarihi	14.01.1974
Doğum yeri	Elazığ
Lise	1987-1990 İzmir Şemikler Lisesi
Lisans	1990-1995 Ege Üniversitesi Mühendislik Fakültesi Bilgisayar Mühendisliği Bölümü

**Çalıştığı Kurumlar**

1995-1997	Göktepe Plastik San. ve Tic. A.Ş.
1997	İZTEK A.Ş.
1997-Devam ediyor	Hava Harp Okulu Komutanlığı Bilgisayar Mühendisliği Bölümü