

**WEB TABANLI ORTAMDA
OLAP ARAÇLARININ KARAR DESTEK
SİSTEMLERİNDE KULLANILMASI**

Aysan ALPAT
Yüksek Lisans Tezi

Bilgisayar Mühendisliği - Bilişim
Kasım 2006

JÜRİ VE ENSTİTÜ ONAYI

Aysan ALPAT'ın "Web Tabanlı Ortamda OLAP Araçlarının Karar Destek Sistemlerinde Kullanılması" başlıklı **Bilgisayar Mühendisliği** Anabilim Dalı **Bilişim** Programındaki, Yüksek Lisans Tezi 18.10.2006 tarihinde, aşağıdaki jüri tarafından Anadolu Üniversitesi Lisansüstü Eğitim-Öğretim ve Sınav Yönetmeliğinin ilgili maddeleri uyarınca değerlendirilerek kabul edilmiştir.

	Adı – Soyadı	İmza
Üye (Tez Danışmanı) :	Prof. Dr. ALİ GÜNEŞ
Üye	:Yard. Doç. Dr. YUSUF OYSAL
Üye	:Yard. Doç. Dr. NİHAL ERGİNEL

Anadolu Üniversitesi Fen Bilimleri Enstitüsü Kurulu'nun
.....tarih ve sayılı kararıyla onaylanmıştır.

Enstitü Müdürü

ÖZET

Yüksek Lisans Tezi

WEB TABANLI ORTAMDA OLAP ARAÇLARININ KARAR DESTEK SİSTEMLERİNDE KULLANILMASI

Aysan ALPAT

Anadolu Üniversitesi

Fen Bilimleri Enstitüsü

Bilgisayar Mühendisliği – Bilişim Anabilim Dalı

Danışman: Prof. Dr. Ali GÜNEŞ

2006, 92 sayfa

İşletmelerin çoğu operasyonel sistemlerden elde edilen işletme verilerini ilişkisel veritabanlarında saklamakta ve bu verileri ilişkisel teknolojiye dayanan analiz araçları ile iki boyutlu olarak analiz etmektedir. İlişkisel veritabanları işletme açısından kritik öneme sahip verileri saklamalarına rağmen geleneksel analiz araçları bu verilerin stratejik bilgiye dönüştürülmesi aşamasında yetersiz kalmaktadır. Ham veriyi işleyecek, anlamlı ve stratejik bilgiye dönüştürecek sistemler gerekli olmaktadır. Bu aşamada OLAP olarak bilinen çok boyutlu analiz sistemleri yöneticilerin karşısına bir çözüm olarak çıkmaktadır.

Yöneticilerin karşılaştığı bir diğer sorun ise geniş bir coğrafyaya yayılmış olan organizasyonlarda teknolojik engellerin azaltılarak kararların zamanında alınmasının sağlanmasıdır. Organizasyonlar karar destek sistemlerine web teknolojilerini entegre ederek bu sorunu çözmektedirler. Web, ağ yapısından ve çok fazla yazılım, kurulum ve bakım maliyeti gerektirmemesinden dolayı birçok organizasyonun istemci-sunucu platformu haline gelmekte ve karar destek yazılımı üreticileri ürünlerine web teknolojilerini entegre etmektedirler. Bu çalışmada OLAP teknolojilerinin işletmeler açısından önemi vurgulanmakta ve OLAP araçlarının web üzerinden kullanım yöntemleri incelenmektedir. Uygulama olarak bir hastanede yer alan ecza deposu ile ilgili veriler çok boyutlu olarak analiz edilmiştir.

Anahtar Kelimeler: Karar Destek Sistemleri, OLAP, Veri Ambarı, Web, Küp

ABSTRACT

Master of Science Thesis

USING OLAP TOOLS ON WEB BASED DECISION SUPPORT SYSTEMS

Aysan ALPAT

Anadolu University

Graduate School of Sciences

Computer Engineering Information Program

Supervisor: Prof. Dr. Ali GÜNEŞ

2006, 92 pages

Most of the enterprises store the data of the enterprise gained from the operational systems in relational databases and these data are analysed as two-dimensional with analysis tools depending on relational technology. Although these databases have very crucial importance data for the enterprises, traditional analysis tools are inefficient at turning the data into strategic information, which can process the raw data and change them into meaningful and strategic information, are needed. At this phase, multi-dimensional analysis systems known as OLAP is a kind of solution for the administrators.

Another problem facing the administrators whose organizations spread on a large geographical zone is to reach a decision in time by reducing the obstacles of the technology. The organizations can solve this problem by integrating the decision systems to web technologies. Because the web does not require much expense for the installation, maintenance and software, it has become client-server platform desired for many organizations and the vendors of decision support software integrate web technologies to their products. In this thesis, significance of OLAP technologies is emphasised in point of view of enterprises and web based access methods of OLAP tools are studied. A group of data, belonging to a pharmaceutical warehouse that is placed in a hospital is analysed in multidimensional aspect in scope of the thesis for implementation.

Keywords: Decision Support Systems, OLAP, Data Warehouse, Web, Cube

İÇİNDEKİLER

	<u>Sayfa</u>
ÖZET	i
ABSTRACT	ii
İÇİNDEKİLER	iii
ŞEKİLLER DİZİNİ	vii
TABLOLAR DİZİNİ	ix
KISALTMALAR DİZİNİ	x
1. GİRİŞ	1
2. KARAR DESTEK SİSTEMLERİ	3
2.1. Karar Destek Sistemi Nedir?.....	3
2.2. KDS'nin Gelişimi.....	3
2.3. Karar Destek Sistemlerinde Geleneksel Teknolojiler.....	5
2.3.1. OLTP sistemleri.....	5
2.3.2. İlişkisel veritabanları.....	6
2.3.3. Elektronik tablolar.....	7
3. OLAP	11
3.1. OLAP Uygulama Türleri.....	13
3.1.1. Analitik raporlama.....	13
3.1.2. Tahminsel analizler.....	14
3.2. Veri Ambarları.....	14
3.3. Çok Boyutlu Yapılar.....	16
3.3.1. Küp bileşenleri.....	19
3.3.1.1. Olay.....	19
3.3.1.2. Boyutlar.....	19
3.3.1.3. Boyut üyeleri.....	20
3.3.1.4. Boyut hiyerarşileri.....	20

	<u>Sayfa</u>
3.3.1.5. Ölçek.....	21
3.4. Boyutlu Modelleme Türleri.....	22
3.4.1. Yıldız (Star) modeli.....	22
3.4.2. Kar tanesi (Snowflake) modeli.....	23
3.5. Temel OLAP İşlemleri.....	24
3.5.1. Slice ve Dice işlemleri.....	24
3.5.2. Drill-down ve Roll-up işlemleri.....	28
3.5.3. Pivot (Rotate) işlemi.....	28
3.5.4. Diğer OLAP işlemleri.....	30
3.6. Codd Kuralları.....	30
3.7. OLAP Modelleri.....	31
3.7.1. Çok boyutlu OLAP (MOLAP).....	32
3.7.1.1. Ön birleştirme.....	33
3.7.1.2. Detaylı bilginin alınması.....	34
3.7.1.3. Seyreklik (Sparsity).....	34
3.7.1.4. MOLAP modelinin avantaj ve dezavantajları.....	34
3.7.2. İlişkisel OLAP (ROLAP).....	35
3.7.2.1. ROLAP modelinin avantaj ve dezavantajları.....	37
3.7.3. MOLAP ve ROLAP modellerinin karşılaştırılması.....	37
3.7.4. Karma OLAP (HOLAP).....	38
3.7.5. Masaüstü OLAP (DOLAP).....	39

4. KARAR DESTEK SİSTEMLERİNDE

WEB TEKNOLOJİLERİ	41
4.1. WEB İşlem Modeli.....	42
4.2. Karar Destek Sistemlerin Kullanılan Web Araçları.....	43
4.3. Web OLAP (WOLAP) Araçları.....	45
4.3.1. Web OLAP yöntemleri.....	46

4.3.1.1. Önceden oluşturulan HTML belgeleri	47
4.3.1.2. Tarayıcıyı ActiveX veya Java bileşenleri ile destekleyerek.....	47
4.3.1.3. Sunucuda OLAP.....	47
4.3.2. Web OLAP yaklaşımının yararları.....	47
5. UYGULAMA	49
5.1. Bir OLAP Uygulaması Olarak SQL Server 2000 Analiz Servisleri.....	49
5.1.1. Microsoft Analiz Servisi'nin özellikleri.....	49
5.1.2. Analiz Servisi mimarisi.....	51
5.1.3. Sunucu mimarisi.....	53
5.1.4. İstemci mimarisi.....	53
5.1.5. Nesne mimarisi.....	54
5.1.6. Analiz Servisi'nde kullanılan araçlar	56
5.1.6.1. Analysis Manager.....	56
5.1.6.2. DTS.....	61
5.2. Örnek OLAP Uygulaması.....	61
5.2.1. Küp boyutlarının oluşturulması.....	62
5.2.1.1. Zaman boyutu.....	62
5.2.1.2. İlaç boyutu.....	65
5.2.1.3. Servis boyutu.....	67
5.2.2. Küp oluşturulması.....	68
5.2.2.1. Depolama türünün belirlenmesi.....	70
5.2.2.2. Oluşturulan küpün işlenmesi.....	71
5.3. İstemci Uygulaması Olarak Cubularity Knowledge Platform Yazılımı.....	73
5.3.1. Knowledge Platform bileşenleri.....	75
5.3.1.1. Knowledge Prospector.....	75
5.3.1.2. Knowledge Server.....	79

	<u>Sayfa</u>
5.3.2. Knowledge Prospector ile analizlerin oluşturulması.....	79
5.3.3. Web yoluyla erişimin sağlanması.....	82
5.3.3.1. Web Renderer kurulumu.....	82
5.3.3.2. Oluşturulan analizlerin web tarayıcıda görüntülenmesi.....	84
6. SONUÇ	86
KAYNAKLAR.....	88

ŞEKİLLER DİZİNİ

2.1. Microsoft Excel elektronik tablosu.....	8
2.2. Microsoft Excel’de grafiksel gösterim.....	8
3.1. İki boyutlu görünüm.....	17
3.2. Üç boyutlu görünüm.....	17
3.3. Üç boyutlu veri küpü.....	18
3.4. Dört boyutlu yapı.....	18
3.5. Zaman boyutunda farklı hiyerarşiler.....	21
3.6. Yıldız modeli.....	23
3.7. Kar tanesi modeli.....	24
3.8. Ürün1’e göre slice işlemi.....	25
3.9. 2001 yılı verisi üzerinde slice işlemi	26
3.10. Dice işlemi ile alt küp oluşturma.....	27
3.11. 2001 yılı verisi üzerinde Drill Down işlemi.....	28
3.12. Pivot işlemi.....	29
3.13. MOLAP mimarisi.....	33
3.14. ROLAP mimarisi.....	36
3.15. HOLAP mimarisi.....	39
4.1. Geleneksel ve Web işlem modelleri.....	43
4.2. Tipik Web OLAP mimarisi.....	45
5.1. Microsoft Analiz Servisi sistem mimarisi.....	52
5.2. İstemci Mimarisi.....	54
5.3. Microsoft SQL Server 2000 Analysis Manager.....	57
5.4. Küp editörü.....	59
5.5. Boyut editörü.....	60
5.6. Eczane veri kaynağı.....	61
5.7. Boyut sihirbazı iletişim kutusu.....	62
5.8. Zaman boyutu için tablo seçimi.....	63
5.9. Boyut tipinin belirlenmesi.....	63
5.10. Zaman boyutunda hiyerarşi yapısı.....	64
5.11. Zaman boyutunda boyut üyeleri.....	64

5.12. İlaç boyutunda tablo seçimi.....	65
5.13. İlaç boyutunda seviyelerin oluşturulması.....	66
5.14. İlaç boyutunda boyut üyeleri.....	66
5.15. Servis boyutunda kullanılan tablolar.....	67
5.16. Servis boyutunda boyut üyeleri.....	67
5.17. Olay tablosunun seçilmesi.....	68
5.18. Ölçeklerin belirlenmesi.....	69
5.19. Eczane küpünün yapısı.....	69
5.20. Depolama türünün belirlenmesi.....	70
5.21. Kümeleme seçeneklerinin ayarlanması.....	71
5.22. OLAP küpünün işlenmesi.....	72
5.23. Eczane küp verisi.....	72
5.24. Knowledge Platform mimarisi.....	74
5.25. OLAP küpü ile bağlantı kurulması.....	80
5.26. Analiz edilecek üyelerin seçimi.....	81
5.27. Tablo biçiminde gösterim.....	81
5.28. Grafiks gösterim.....	82
5.29. Alt web dizininin oluşturulması.....	83
5.30. Web tarayıcıda analizlerin görüntülenmesi.....	84
5.31. 2000-2001_Acil_Dahiliye_Net_Gelir.c3 analiz dosyası.....	85

TABLÖLAR DİZİNİ

2.1. Karar destek sistemlerinin gelişimi.....	5
3.1. Operasyonel sistemler ve veri ambarı sistemlerinin karşılaştırılması.....	16
3.2. MOLAP ve ROLAP modellerinin karşılaştırılması.....	38

KISALTMALAR DİZİNİ

ADO MD	:ActiveX Data Objects Multidimensional
API	:Application Program Interface
ASP	:Active Server Pages
CGI	:Common Gateway Interface
DSO	:Decision Support Objects
DTS	:Data Transformation Services
EIS	:Executive Information Systems
ETL	:Extract Transform Load
GDSS	:Group Decision Support Systems
HOLAP	:Hybrid Online Analytical Processing
HTML	:HyperText Markup Language
HTTP	:HyperText Transfer Protocol
IIS	:Internet Information Services
KDS	:Karar Destek Sistemi
MDDB	:Multidimensional Databases
MDX	:Mutidimensional Extensions
MIS	:Management Information Systems
MMC	:Microsoft Management Console
MOLAP	:Multidimensional Online Analytical Processing
ODBC	:Open Database Connectivity
OLAP	:Online Analytical Processing
OLE DB	:Object Linking and Embedding Database
OLTP	:Online Transactional Processing
PTS	:PivotTable Service
RDBMS	:Relational Database Management Systems
ROLAP	:Relational Online Analytical Processing
SQL	:Structured Query Language
URL	:Universal Resource Locator
XML	:Extensible Markup Language

1. GİRİŞ

Bilgi tüm başarılı kararların temelinde yer almaktadır. Özellikle rekabet ortamının şiddetlendiği günümüzde yöneticiler karar alma sürecinde bilginin işletmelerine kazandırdığı avantajları her geçen gün daha iyi anlamaktadır. Bu rekabetçi ortamda bilgi teknolojilerine yapılan yatırımlar işletmelerin stratejik hedeflerinin gerçekleştirilmesinde büyük önem kazanmakta ve pazar içindeki konumlarının belirlenmesinde kritik bir rol üstlenmektedir.

Yöneticiler karar sürecinde eskiye oranla daha fazla veri yığınıyla karşı karşıya kalmaktadır. Geçmişte odak noktası işletme verilerinin toplanması ile sınırlıyken gelişen teknolojiler sayesinde odak noktası toplanan verilerden anlamlı sonuçlar çıkartılması olarak değişmiştir. Veri miktarı düşük olduğunda bu verilerin anlaşılması ve yorumlanması sorun olmamaktadır, ancak veri miktarı arttığında karar vericilerin bu verileri anlamaları ve yorumlamaları zorlaşmaktadır. Yöneticiler, bu veri yığınları arasından kendileri için gerekli olan bilgileri çıkartıp görüş açlarına dahil ederek veriyi anlamlı bir bütün olarak gösterecek sistemlere ihtiyaç duymaktadırlar. Günümüzde kullanılan ilişkisel teknolojiye ve verinin iki boyutlu görünümüne dayanan geleneksel karar destek sistemi araçları bu ihtiyacı karşılamaktan uzaktır.

İş ihtiyaçları çok boyutlu analiz gerektirmektedir. Bu noktada anlamsız görünen veri yığınlarının düzenlenmesi ve çok boyutlu olarak analiz edilmesinde veri ambarları ve OLAP (online analytical processing) araçları olarak bilinen teknolojiler, organizasyonların karşısına bir çözüm olarak çıkmaktadır.

Bir veri ambarı, operasyonel sistemler gibi dahili ve harici kaynaklardan veri toplayan, bu verileri seçip düzenleyerek çok boyutlu analize uygun hale getiren birleştirilmiş bir depo olarak karar destek ve analiz araçları için temel oluşturmaktadır. Veri ambarlarında düzenlenerek analize uygun hale getirilen veriler, OLAP araçları olarak isimlendirilen çok boyutlu analiz araçları kullanılarak, işletme açısından değer ifade eden stratejik bilgiye dönüştürülmektedir. OLAP araçları veri ambarında bulunan, belirli bir konuyla ilişkili veriler üzerinde analitik işlemlerin yapılmasına izin vererek, karar vericilerin veriye bir bütün olarak ve istenilen açıdan bakabilmelerine izin

vermektedir. Bu tip çok boyutlu analiz araçları ile ham veriler anlamlı, stratejik bilgiye dönüştürülmektedir.

Son yıllarda karar destek sistemlerinde kullanımı yaygınlaşan bir diğer teknoloji ise web teknolojileridir. Web teknolojileri ile organizasyon içinde veya dışında kullanıcıların farklı yerlerde bulunan verilere ve karar destek sistemi araçlarına bir tarayıcı yardımıyla ulaşabilmeleri mümkün olmaktadır. Bu şekilde ihtiyaç duyulan verilere hızlı ve zamanında erişim, organizasyonlarda kararların zamanında alınmasına yardım ederek verimliliği ve alınan kararların kalitesini yükseltmektedir.

2. KARAR DESTEK SİSTEMLERİ

2.1. Karar Destek Sistemi Nedir?

Karar Destek Sistemi (KDS) kavramı ortaya çıkışından itibaren birçok farklı şekilde tanımlanmıştır. KDS, problemlerin tanımlanması ve çözülmesi, karar süreç işlerinin tamamlanması ve kararlar verilmesi amacıyla iletişim teknolojileri, veri, belgeler, bilgi ve modeller kullanarak karar vericilere yardımcı olmak amacıyla tasarlanmış bilgisayar temelli sistem veya alt sistemlerdir [1].

Basit anlamda bir KDS, çok sayıda olay ve yöntem girdilerini, karar vericilerin karar verme yeteneklerini arttırabilen ve kolaylaştırabilen anlamlı karşılaştırmalar ve grafiklere dönüştürmeyi olanaklı kılan ve üstlenen bir bilgisayar yazılımıdır. KDS, karar vericiye, bilginin yararlı bir şekilde işlenmesinde, değerlendirilmesinde, sınıflandırılmasında ve düzenlenmesinde yardımcı olmaktadır [2].

Genel olarak bir KDS'ni aşağıdaki gibi özetlemek mümkündür [3]:

- KDS karar vericilerin işlerini yapması için değil, onlara karar vermede destek olması amacıyla kullanılır.
- Yarı yapılandırılmış ve yapılandırılmış karar ortamlarında destek sağlar.
- Karar verme işleminin bütün evrelerini destekler.
- En üst düzeyden en alt düzeye kadar tüm yönetim seviyelerini destekler.
- Etkileşimli ve kullanıcı dostudur.
- Temel olarak veri ve modeller kullanmaktadır.

2.2. KDS'nin Gelişimi

KDS'nin kökleri yöneticilerin örgütsel ortamlarda karşılaştıkları günlük problem ve kararların sayısal modeller şeklinde ifade edilmesinden gelişmiştir. Kavram 1970'lerden önce doğmuştur [3].

1965 öncesinde yüksek ölçekli bilgi sistemleri kurmak oldukça maliyetliydi. Bu zamanlarda IBM 360'lar ve diğer daha güçlü mainframe sistemler geliştirilmesiyle büyük firmalarda MIS'in gelişimine kadar daha pratik çözümler bulunmuş ve maliyetler daha düşük seviyelere çekilmiştir. 1960'ların sonlarına doğru uygulanabilir-model temelli KDS ya da yönetim karar sistemi bilgi sisteminin yeni bir tipi olarak ortaya çıkmıştır.

1968-1969'da Scott Morton, bilgisayarlar ve analitik modellerin, yöneticilerin kritik kararlar almasında nasıl yardımcı olabileceği konusunda çalışmıştır. Bu dönemde Morton, çamaşırhane ekipmanları için ürün planlamasının yapılmasında genellikle pazarlama ve ürün yöneticilerinin kullandığı bir yönetim karar sisteminin (MDS) kullanıldığı bir deneyi yönetmiştir.

1974'de Gordon Davis MIS kavramını "Bir organizasyonda operasyonları, yönetimi ve karar verme fonksiyonlarını desteklemek için bilgi sağlama amaçlı bütünleşik, insan-makine sistemi" olarak tanımlamıştır.

1975 ile birlikte Little, bilgisayar temelli modellemenin sınırlarını genişletmiştir. Little, Brandaid isimli KDS'ni ürün, promosyon, fiyatlandırma ve reklam kararlarını desteklemek amacıyla tasarlamıştır. Little, yönetsel karar vermeye destekte model ve sistem tasarımı için dört önemli kriter tanımlamıştır: Sağlamlık, kolay kontrol, basitlik ve ilgili ayrıntılarda bütünlük.

1980'lerin ortalarında grup halinde alınan kararları destekleyen grup karar destek sistemleri (GDSS) ve üst düzey yöneticileri stratejik kararlarda destekleyen üst düzey bilgi sistemleri (EIS) ortaya çıkmıştır [4].

1990'ların başlarında ise KDS kurulumu için 4 güçlü araç ortaya çıkmıştır. KDS için ilk yeni araç veri ambarı idi. Veri ambarlarının çıkışını takiben iki yeni araç: OLAP ve veri madenciliği ortaya çıkmıştır. Dördüncü yeni araç grubu ise web ile ilişkili teknolojidir [5]. Tablo 2.1. de karar destek sistemlerinin günümüze kadar ki gelişimi gösterilmektedir.

Tablo 2.1. Karar Destek Sistemlerinin Gelişimi [4]

1960	1970	1980	1990
MIS ve Yapısal Raporlar	Brandaid	GDSS	İş Zekası
Etkileşimli Sistem Araştırmaları	MDS	EIS	Veri Ambarları
Teori Geliştirme		Uzman Sistemler	Veri Madenciliği
			OLAP
			Portallar

2.3. Karar Destek Sistemlerinde Geleneksel Teknolojiler

Günümüzde çoğu işletmeler kayıt ekleme, silme, güncelleme gibi OLTP (online transactional processing) tipi işlemler sonucu elde edilen verileri genellikle ilişkisel veritabanlarında tutmaktadır. İlişkisel veritabanı sistemlerinde tutulan bu veriler, geleneksel sorgulama araçları kullanılarak 2 boyutlu olarak analiz edilmektedir.

2.3.1. OLTP sistemleri

Kayıt ekleme, silme, güncelleme gibi günlük yapılan operasyonel işlemlerde kullanılan sistemlere OLTP sistemleri adı verilmektedir. OLTP sistemlerinden elde edilen veriler genellikle ilişkisel veritabanlarında saklanmaktadır. Bu tip sistemlerde yapılan işlemlere örnek olarak stok giriş-çıkışı yapma, bunları güncelleme gibi basit ve küçük işlemleri vermek mümkündür.

OLTP sistemleri bir organizasyondaki işlerin büyük bölümünü oluşturan günlük işlemleri yerine getirerek bilgi sistemlerinin çekirdeğini oluşturmaktadır [6].

Bu tip geleneksel operasyonel sistemler verinin hızlı, doğru ve verimli bir şekilde veritabanına alınmasında oldukça iyi olmalarına rağmen verilerin anlaşılabilir analizlerinin yapılabilmesinde yeterli olamamaktadır [7]. Stratejik

kararlarda kullanılmak üzere gerekli özet bilgileri sağlamaktan ziyade organizasyonun o andaki durumuyla ilgili anlık veriler vermektedir. İşlemler sürekli devam etmekte, veriler ve veriler arasındaki ilişkiler her an değişmekte ve güncellenmektedir [8].

2.3.2. İlişkisel veritabanları

Bu tip veritabanları veriyi seçme, depolama ve SQL (Structured Query Language) yardımıyla basit sorgulamalar yapmak için uygundur. Satış ayrıntıları, ürün , bölge, fiyat, müşteri benzeri. veriler doğru ve hızlı bir şekilde alınır. Bu veritabanları terabyte büyüklüğünde detay veriyi depolayabilmektedirler. İlişkisel veritabanlarında veriler satır ve sütunlardan oluşan birbirleriyle ilişkili tablolar halinde tutulmaktadır.

İlişkisel veritabanları işletmeyle ilgili kritik sayılabilecek verileri saklamalarına rağmen bu verileri stratejik bilgiye dönüştürmek için gerekli olan esnek ve zamana bağlı analitik sorgular yapılmasında yetersiz kalmaktadırlar. Bu tür sistemlere SQL yardımıyla “Ürün x’in stoktaki durumu nedir?” ya da “ Ürün x’in Ocak ayındaki satışları ne kadar olmuştur?” gibi basit işlemsel sorgular göndererek yanıtlar alabilmek mümkündür. Fakat bu tür sorgular karar almada yöneticiye stratejik bilgiler sağlamaktan uzaktır. “Ürün x’in ve ürün y’nin ocak ayında Karadeniz Bölgesi’ndeki satışları, geçen yılın aynı dönemdeki satışlarına göre ne kadar değişim göstermiştir?” gibi daha karmaşık ve analitik bir sorgu gönderildiğinde buna cevap alınması oldukça zor olacaktır.

İlişkisel veritabanlarının dezavantajlarını şu şekilde özetlemek mümkündür [9]:

- İş ihtiyaçları çok boyutlu veri gösterimini gerektirmesine rağmen dinamik değildir ve iki boyutludur.
- Esnek değildir. Geçmiş dönemlere ait verilere değil şimdiki veriye uygundur.
- Raporların elde edilmesi için uzman personele ihtiyaç duyulmaktadır. Dolayısıyla veritabanı kullanımı son kullanıcıya uygun değildir.
- Büyük miktarlarda veri olduğunda rapor üretmek zaman almaktadır.

2.3.3. Elektronik tablolar

Elektronik tablolar karar vericilerin iş analizlerinde yaygın olarak kullandığı önemli araçlardan biridir. Bu araçlar iş dünyasında muhasebe, insan kaynakları, bütçe analizleri, finansal analizler, istatistiksel analizler ve “what-if” analizlerinde sıklıkla kullanılmaktadır [9].

Bir elektronik tablo insanla bilgisayar arasındaki etkileşimi arttıran bir uygulamadır. Bu tip araçlar birbirinden ayrı olan verileri formüller yardımıyla hızlı bir şekilde birleştirme ve düzenleme imkanı sağlamaktadır. Aylara göre toplam satış miktarı ve satış tutarı gibi verilerden formüller kullanarak aylık kazancın belirlenmesi ve verilerin grafiksel olarak gösterilmesi gibi özellikler analistlerin konuyla ilgili fikir sahibi olmalarını ve konuya değişik açılardan bakarak yeni anlayışlar kazanmalarına imkan sağlamaktadır. Karar verici, problemle ilgili kafasında tasarladığı durumları bir elektronik tabloya aktarabilmekte ve hücrelerde yapacağı olası değişikliklerle durumdaki herhangi bir değişikliğin sonuca nasıl yansıtacağını görebilmektedir [10].

Elektronik tablolar satır ve sütunlardan oluşmaktadır. Bu satır ve sütunlar yardımıyla iki boyutlu analiz gerçekleştirilmektedir. Satır ve sütunlar kesişerek verilerin girileceği hücreleri oluşturmaktadır. Her çalışma sayfasında verilerin girilebileceği yüzlerce hücre bulunmaktadır. Her hücre tek bir adrese karşılık gelmektedir. Elektronik tablolarda genellikle sütunlar harflerle satırlar ise numaralarla ifade edilmektedir. Hücreler ise buldukları satır ve sütun değerleriyle ifade edilmektedir. Örnek verilecek olursa sütun B ve satır 6’da bulunan hücre, hücre B6 olarak isimlendirilmektedir.

Günümüzde Microsoft Excel ve IBM Lotus 1-2-3 iş dünyasında yaygın olarak kullanılan elektronik tablo yazılımlarındandır [11].

Şekil 2.1. de görüldüğü gibi girdi ya da çıktı sütunlarında yapılacak herhangi bir değişiklik sonuca direkt olarak yansıtacaktır. Şekil 2.2. de ise yapılan analizin grafiksel gösterimi görülmektedir.

Microsoft Excel

Dosya Düzen Görünüm Ekle Biçim Araçlar Veri Pencere Yardım

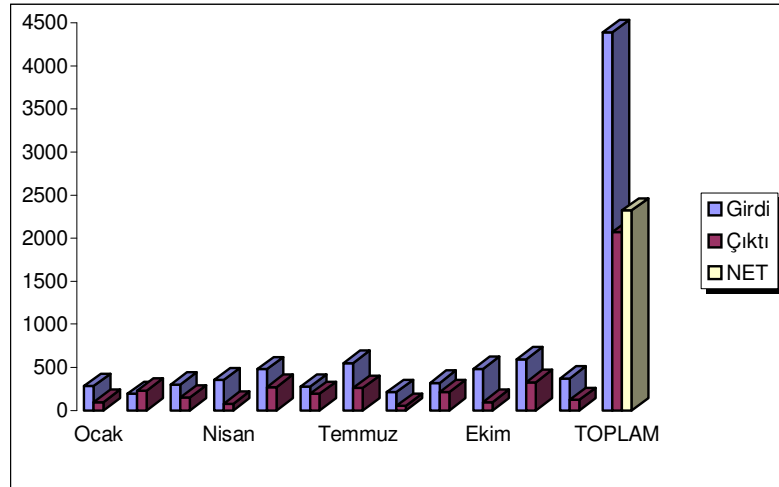
A14 = TOPLAM

Kitap1

	A	B	C	D	E	F	G	H
1		Girdi	Çıktı	NET				
2	Ocak	280	95					
3	Şubat	190	230					
4	Mart	300	148					
5	Nisan	350	75					
6	Mayıs	475	270					
7	Haziran	274	190					
8	Temmuz	550	260					
9	Ağustos	210	50					
10	Eylül	320	210					
11	Ekim	480	95					
12	Kasım	590	325					
13	Aralık	370	120					
14	TOPLAM	4389	2068	2321				
15								
16								
17								

Hazır SAYI

Şekil 2.1. Microsoft Excel elektronik tablosu



Şekil 2.2. Microsoft Excel'de grafiksel gösterim

Elektronik tabloların tüm bu yararlarına rağmen birçok kusurları da mevcuttur. İş verileri arttıkça elektronik tablodaki çalışma sayfaları da çoğalmaktadır. Örneğin bir imalat firmasında kullanılan üretim ve raporlama sisteminde analist, her bir ürün için farklı bölümlerdeki ve farklı yerlerdeki çıktılar yılın belli dönemlerine göre karşılaştırmak istediğinde, birçok çalışma sayfasıyla bağlantı kurmak zorunda kalacaktır. Bu tip çoklu çalışma sayfası sistemi, yeni ürün tiplerini eklemek isteyen veya imalat sürecinde sadece bir ürünün maliyetini değiştirmek isteyen analistler için ciddi sorunlar doğurmaktadır.

Çalışma sayfaları, verinin zamanın belli bir anındaki statik görüntüsünü vermektedir. Bu çalışma sayfalarına yeni satır ve sütunlar eklemek kolay olmayacaktır. Aynı verinin değişik görünümünü elde etmek kolay olmamaktadır. Çünkü çalışma sayfaları veri ve formülleri her hücre için özel olarak kaydetmektedir. Aynı zamanda büyük miktarlarda veriyle çalışıldığında hesaplama yavaş olmaktadır.

İyi bilinen bir diğer çalışma sayfası sorunu ise organizasyon çapında yapılan geniş bir planlamayı ve istenilen raporlama şeklini bir çok kez sınırlandırıyor olmasıdır. Şu örnek verilecek olursa: Bir analist, birbiriyle bağlantılı bir grup çalışma sayfasından birini kopyalar ve bunu meslektaşına gönderir. Meslektaşı bir çok değeri güncelleyerek bunu üçüncü bir kişiye değerlerini girmesi için gönderir. Değerler değiştirilir ve çalışma sayfası kaynağına geri gönderilir. Çalışma sayfası geri gönderildiğinde analist, tüm sistemi güncellemek için bağlantılı çalışma sayfalarına değiştirilmiş rakamları tekrar girmek zorunda kalacaktır [9].

Elektronik tabloların dezavantajlarını özetle şu şekilde sıralamak mümkündür [9]:

- Çoklu çalışma sayfaları tüm iş analiz ihtiyaçlarını karşılamak durumundadır.
- Bilgiye erişim ve şablonların muhafaza edilmesi zordur.
- Hesaplamaları ve çalışma sayfaları arasındaki bağlantıları ortadan kaldırmadan, yeni veri alanları (satır ve sütunlar) eklemeye yetersizdir. Yapı biçimlendirilemez, görünüm değiştirilemez.

- Yavaş ve yetersiz hesaplama zamanı.
- Çoklu kullanıcılar olduğunda bilgi paylaşımı ve güncellenmesi zordur.

3. OLAP

Yöneticiler bilgiyi ve bu bilgiyi kullanmak için gerekli olan sistemleri rekabetçi ortamda avantaj sağlamak için kritik bir kaynak olarak görmektedirler. OLAP veya çok boyutlu veri analizi, organizasyonel bilgiyi en iyi şekilde kullanmanın bir yoludur. OLAP yazılımları işletme veritabanlarından veriyi alarak özetlenmiş bilgiyi sunmaktadırlar [12]. Çevrimiçi analitik işleme anlamına gelen OLAP terimi ilk olarak 1993 yılında, ilişkisel veritabanı kavramının da çerçevesini çizen Dr. E. F. Codd tarafından kullanılmıştır.

OLAP Council (OLAP Konsorsiyumu) tarafından OLAP, kullanıcıların anlayabileceği biçimde işletmenin gerçek boyutunu yansıtacak ham veriden dönüştürülen stratejik bilginin çok çeşitli olası görünümüne analist, yönetici ve uzmanların hızlı, tutarlı ve etkileşimli erişimine izin veren bir yazılım teknolojisi sınıfı olarak tanımlanmıştır [1].

OLAP, çok boyutlu olarak düzenlenmiş özet veriden yararlanarak karar vermeye yardımcı olmaya çalışmaktadır. Geleneksel veritabanları genellikle operasyonel görevlerle ilgili olan OLTP sistemlerini desteklemek amacıyla geliştirilmişlerdir. Geleneksel veritabanı yönetim sistemleri çok boyutlu analizde gerekli olan sentezleme, analiz ve konsolidasyon için güçlü fonksiyonları sağlamaktan uzaktırlar [13,14]

Günlük yapılan işlemlerde kullanılan OLTP sistemleri ile yönetsel karar vermeyi destekleyen OLAP sistemleri arasındaki farkı anlamak önemlidir. Yöneticiler geleneksel yönetim bilgi sistemleri kullanılarak önceden hazırlanmış halde bulunan raporları nadiren okumak isterler. Genellikle bilgiyi aramak veya organizasyonda neler olduğu hakkında bir fikir sahibi olmak amacıyla kullanımı kolay ve etkileşimli bir sistemi tercih etmektedirler. Böyle bir sistem OLAP tarafından sağlanabilmektedir [12].

Kullanıcılara veriyi çok boyutlu olarak inceleme ve analiz etme imkanı sağlayan OLAP sayesinde karar vericiler işle ilgili belirli bir dönemi kapsayan birçok farklı boyut arasında düzenlemeler, karşılaştırmalar ve trendlere dayanan sorgular yapabilmektedirler [7].

Analistlere, büyük miktarlardaki veriye özet seviyesinde hızlı erişim sağlayabilme ve iş planlamasına destek olmak amacıyla veri üzerinde dikkatli ve

esnek analizleri gerçekleştirebilme olanakları sağlayan OLAP yazılımları, veri düzenleme, what-if sorguları ve trend analizlerinin yapılabileceği etkili analiz araçlarından oluşmaktadır [15]. Bu analiz araçları kullanıcılara veri üzerinde dilimleme (slice ve dice) işlemleri, ayrıntılı veriye inme (drill down) işlemleri yaparak çok boyutlu analizler gerçekleştirme imkanı vermekte ve stratejik kontrol evresini kolaylaştırmaktadır. Güncel miktarları karşılaştırma olanağı vererek sonuçlardaki değişiklikleri daha iyi anlamaya yardım etmektedir [16].

Bazen OLAP ve veri ambarlama eş anlamlı olarak kullanılmaktadır. Veri ambarı ve OLAP kavramlarının birbirlerinin tamamlayıcısı olduklarını söylemek mümkündür. Veri ambarları ham veriyi son kullanıcılara uygun hale getirip tutarlılığını ve tamlığını temin ederken, OLAP son kullanıcıların analitik ihtiyaçları üzerine odaklanmaktadır. Veri ambarları, OLAP uygulamasına durum değişikliğini yansıtabilmek için özet veriyi güncellemekten sorumludur [14,17,18].

Bir organizasyonda OLAP'ın rolü, yönetimi veya karar verme işlemini desteklemek amacıyla analitik kaynaklara kolay bir şekilde etkileşimli erişimi sağlamaktır. Karar destek kuramına göre analitik kaynaklar iki türe ayrılmaktadır: veri (statik bilgi) ve modeller (dinamik bilgi) [19]. OLAP bu tanıma mükemmel şekilde uymaktadır. Ham veri ve türetilmiş veri üzerinde sorgulara yanıt veren bir bilgi sunucusudur. Ham veri, veritabanına yüklenen veridir, türetilmiş veri ise ham verinin bir özeti, bir what-if senaryosundan ya da bağımsız olarak hesaplanarak türetilmiş bir tahmin olabilir. OLAP, erişim ve hesaplama kolaylıklarını birleştiren bir teknoloji olarak diğerlerine oranla üstündür [20].

OLAP tanımını beş kritik kelimeyle özetlemek mümkündür: Paylaşılan çok boyutlu bilginin hızlı analizi (Fast Analysis of Shared Multidimensional Information veya kısaca FASMI) [21].

- Fast: Sistem tarafından çoğu yanıtların kullanıcılara yaklaşık beş saniye içerisinde dağıtılması hedeflenmektedir.
- Analysis: Sistemin herhangi bir iş mantığı ve istatistiksel analizin üstesinden gelebilmesidir.

- Shared: Sistem gizlilik için tüm güvenlik gereksinimlerini yerine getirmektedir. Çoklu yazma erişimi gerektiğinde eşzamanlı güncelleme uygun kademedeki kilitlenmelidir.
- Multidimensional: En önemli gereksinimdir. Sistem çok boyutlu kavramsal gösterime izin vermelidir.
- Information: Tüm veri ve türetilmiş bilginin gerekliliğidir. Önemli olan ne kadar verinin depolandığı değil girilen verinin ne kadarının kullanılabilirliğidir.

OLAP sistemlerinin sağladığı olanaklardan bazılarını şu şekilde sıralamak mümkündür [15]:

- Çok çeşitli kullanıcı grupları yararlanabilir(üst düzey yöneticiler, analistler).
- Ad Hoc veri düzenleme, analiz ve raporlamaya izin verir.
- Özel indeksleme teknikleri kullanarak özet veriye hızlı erişim sağlar.
- Verinin çok boyutlu görüntüleri kullanılır.
- Güçlü karar destek sistemleri için temel teşkil eder.

3.1. OLAP Uygulama Türleri

İş analizine destek olan bu uygulamaları dört ana grupta incelemek mümkündür [22]:

- Standart raporlama
- Ad-hoc sorgu ve raporlama
- Çok boyutlu analitik raporlama
- Tahmin analiz ve planlama

3.1.1. Analitik raporlama

Analitik uygulamalar farklı iş çeşitlerini desteklemektedir.

- Muhasebe: Tahmin, bütçeleme, maliyet ve karlılık analizleri
- İnsan kaynakları: İş çizelgeleme ve optimizasyon
- Dağıtım: Çizelgeleme ve optimizasyon
- Satış gücü otomasyonu: Çapraz satış ve bölge analizi
- Pazarlama: Pazar bazlı analizler.
- Perakende satış: Bölge ve demografik analizler

- Sağlık hizmetleri: Sonuç analizleri
- Finansal servisler: Risk değerlendirme ve yönetimi

3.1.2. Tahminsel analizler

Planlama uygulamaları organizasyonların sonuçları tahmin etmesine izin vermektedir. Modeller ve senaryo yönetimi gibi tahminsel analiz araçları kullanılarak yeni veriler üretilmektedir. Bu tip uygulamalara örnek olarak kurumsal bütçeleme ve finansal analizleri verebilmek mümkündür.

Bütçeleme ve finansal analiz sistemleri organizasyonlara geçmiş performanslarını analiz etme, kazanç hedeflerine yönelik yönetim ve finansal plan üzerinde yapılacak değişikliklerin sonuçlarının modellenmesi imkanlarını vermektedir. Bu şekilde yönetimin harcama ve yatırım kararlarını doğru bir şekilde alması sağlanmaktadır. Finans analistleri fiyatlardaki dalgalanmalar gibi beklenmedik etmenlere göre alternatif bütçeleme ve yatırım planları hazırlayabilmektedirler.

3.2. Veri Ambarları

Karar verme sürecinde zamanında sağlanan bilginin rekabet ortamındaki avantajının daha iyi anlaşılmasına başlanmasıyla birlikte, son yıllarda veri analizinin önemi giderek daha da artmaktadır. Bir veri ambarı, özellikle analitik amaçlar için düzenlenmiş, birleştirilmiş tarihsel veriyi saklayan büyük bir veri havuzu olması sebebiyle bu süreçte kullanılmaktadır [23].

Bir veri ambarını temel olarak operasyonel veritabanlarından ayıran iki önemli fark bulunmaktadır. Bunlardan ilki, veri ambarının önceden varolan veritabanı kaynaklarından sağlanan bilgiyi bütünleştirmesidir. İkinci önemli fark ise veri ambarlarının OLAP gibi analitik işlemler için kullanılmasıdır [24].

Operasyonel veriler standart SQL dili ile sorgulanırken, veri ambarları genişletilmiş SQL ile sorgulanmaktadır. SQL'deki bu genişletmeler çok boyutlu genişletmeler olarak isimlendirilmektedir ve MDX olarak bilinen çok boyutlu sorgulama dilinin ortaya çıkmasına neden olmuştur [25].

W. H. Inmon [26] veri ambarını yönetimin karar verme sürecine yardımcı olan konu temelli, bütünleşik, zaman değişkeni olan ve zamanla değişmeyen veri yığını olarak tanımlamıştır.

Bu özellikleri şu şekilde açıklamak mümkündür:

- **Konu temelli:** Veri ambarı organizasyonun esas konuları üzerine odaklanır. Örneğin müşteriler, ürünler veya satışlar gibi. Karar vericiler için verinin modellenmesi ve analizini amaç edinir. OLTP işlemleri gibi günlük işlemler üzerine odaklanmamaktadır.

- **Bütünleşik:** Veri tutarlı bir biçimde depolanır. Veriler ilişkisel veritabanları ve OLTP sistemleri gibi çeşitli veri kaynaklarından alınır. Veri ayıklama ve entegrasyon teknikleri uygulanır. İsimlendirmeler, nitelik ölçütleri ve kodlama yapıları herkesçe kabul görülen şekilde yapılır. Örneğin otel konaklama bedeli fiyat, vergi, kahvaltının dahil olup olmaması gibi bileşenlerden oluşur. Veri, veri ambarına alındığında bu dönüştürülmeler yapılmış olur.

- **Zaman değişkenli:** Veri belirli bir zamanla ilişkilendirilir (dönem, mali yıl, ödeme periyodu v.b.) ve sistemlerde toplanmış veriler operasyonel sistemlerde olduğundan daha uzun zaman dilimlerini kapsayan periyotlarla veri ambarına aktarılır.

- **Zamanla değişmeyen veri:** Veri, veri ambarına alındığında bir daha değiştirilmez. Veri üzerinde ekleme, silme, güncelleme gibi işlemler yapılmaz. Hareket işleme (transaction processing), geri alma ve aynı anda kullanım yöntemleri gerektirmez. Veri erişimi konusunda yalnızca iki işlem gerektirmektedir: Verinin ilk yüklenmesi ve sonradan veriye erişim [26,27,28].

Veri ambarı kavramı artan rekabetçi ortamın iş verisini hızlı analiz etme ihtiyacından doğmuştur. Varolan operasyonel sistemler bu ihtiyacı aşağıdaki nedenlerden ötürü karşılayamamaktadır [29]:

- Çevrimiçi tarihsel veriden yoksundur.
- Analiz için gerekli veri farklı operasyonel sistemlere yerleşmiştir.
- Operasyonel sistemlerdeki sorgu performansı oldukça düşüktür
- Operasyonel veritabanı tasarımları karar destek için uygun değildir.

Tablo 3.1. de operasyonel sistemler ve veri ambarı sistemleri arasındaki farklar gösterilmiştir.

Tablo 3.1. Operasyonel sistemler ve veri ambarı sistemlerinin karşılaştırılması [30].

Veri Ambarı Sistemleri	Operasyonel Sistemler
Yönetim tarafından kullanılır	Alt kademedeki çalışanlar tarafından kullanılır
Stratejik değerdedir	Taktiksel değerdedir
Stratejik işlemleri destekler	Günlük işlemleri destekler
Çevrimiçi analizler için kullanılır	Hareket işleme için kullanılır
Geçmişe dayalı veri depolanır	Şimdiki veri depolanır
Tahmin edilemeyen sorgu deseni	Tahmin edilebilir sorgu deseni

3.3. Çok Boyutlu Yapılar

Çok boyutlu yapılar ya da görünümler her kenarı bir boyutu temsil eden veri küpleri olarak düşünülebilir. Her boyut bir analiz bileşenini göstermektedir. Örneğin bir ürün sınıfı, maliyet tipi ya da zaman olabilir. Çok boyutlu yapıda her hücre, tüm boyutlarla ilişkili özet veri içermektedir [31].

Bu çok boyutlu yapı verinin karmaşık analizine izin vermektedir. Üç ya da daha fazla boyutta görüntülenen verinin analiz yetenekleri, iki boyutlu verinin analiz yeteneklerine göre daha fazla olacağından, çok boyutlu veri kaynakları yani küpler giderek daha fazla tercih edilir duruma gelmektedir. Bir küpte her kenar yani her boyut aynı sayıda üye içermek zorunda değildir. Fakat tüm bu yapılar genel olarak küp olarak isimlendirilmektedir.

Şekil 3.1.'de yıllara göre düzenlenmiş ürün satış verisi görülmektedir. Bu yapı, zaman ve ürün olmak üzere iki boyuttan oluşan bir görünümdür. Bu yapıyı

herhangi bir elektronik tablo yazılımıyla kolaylıkla elde etmemiz mümkün olmaktadır. Zaman ve ürün boyutuna ek olarak bir de bölge boyutu gibi bir boyut eklenmek istenebilir. Üçüncü boyut olarak bölge boyutu da bu yapıya eklendiğinde yapı Şekil 3.2.'deki gibi olacaktır.

Şekil 3.2. iki boyutlu tablolardan oluşturulmuş üç boyutlu bir yapıyı göstermektedir. Bu yapıyı Şekil 3.3.'te olduğu gibi bir küp olarak düşünmek mümkündür. Şekil 3.4. de ise küplerden oluşan dört boyutlu yapı görülmektedir.

Ürün Boyutu

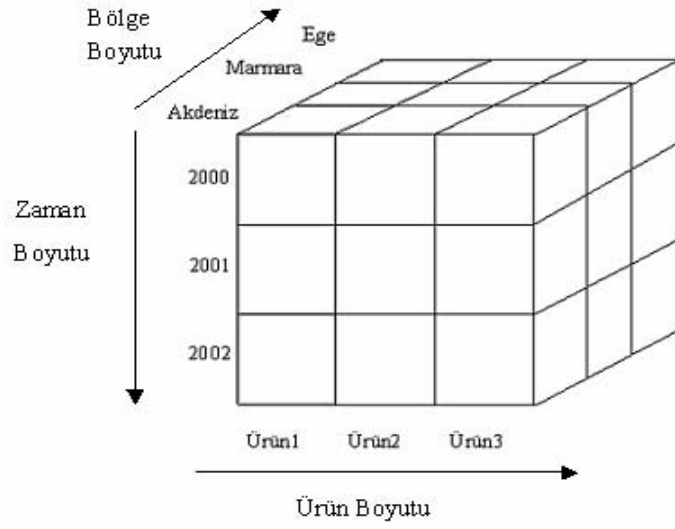
→

		Ürün 1	Ürün 2	Ürün 3
Zaman Boyutu	2000	4800	2000	1700
	2001	2300	500	980
	2002	1900	3800	2100

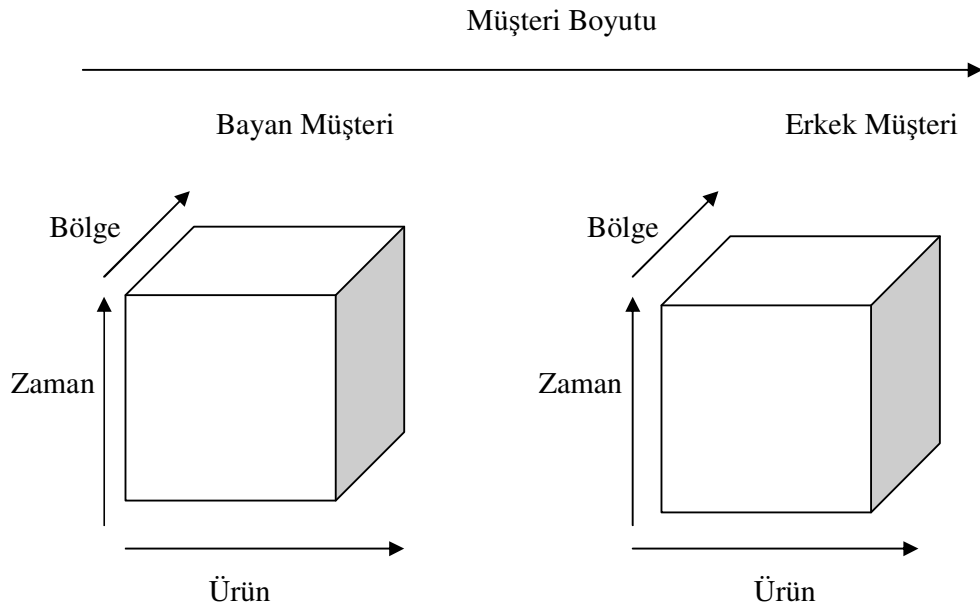
Şekil 3.1. İki Boyutlu Görünüm

	Akdeniz			Marmara			Ege		
	Ürün1	Ürün2	Ürün3	Ürün1	Ürün2	Ürün3	Ürün1	Ürün2	Ürün3
2000	1300	900	550	3000	500	400	500	600	750
2001	400	250	230	800	150	300	1100	100	450
2002	700	2250	900	500	800	850	700	750	350

Şekil 3.2. Üç Boyutlu Görünüm



Şekil 3.3. Üç boyutlu veri küpü



Şekil 3.4. Dört Boyutlu Yapı

Şekil 3.3.'ten de anlaşılacağı gibi 2002 yılında Marmara Bölgesi'nde gerçekleştirilen satışların ne kadarının Ürün1'e ait olduğu öğrenilmek istendiğinde bu sorunun cevabı kolaylıkla alınabilmektedir. Boyut sayısı artırılarak iş verisi üzerinde daha derin analizler yapmak mümkündür. Dördüncü bir boyut eklendiğinde ortaya Şekil 3.4'teki gibi 3 boyutlu küplerden oluşan bir yapı çıkmaktadır.

Bölge, zaman ve ürün boyutlarının yanısıra yapıya satış yapılan müşterilerin cinsiyetleri de dahil edilerek Şekil 3.4.'te görüldüğü gibi 4 boyutlu bir yapı elde edilerek iş verisinin müşterilerin cinsiyeti açısından analiz edilmesi de mümkün olmaktadır [18,28,32].

Ürün ve zaman boyutlarından oluşan iki boyutlu yapıda veriler özet seviyesindedir. Bu verilere tüm bölgelerde yapılan satışlar ve tüm müşteriler dahil olmaktadır. Bu iki boyutlu veri yapısının stratejik kararların alınmasında karar vericiye ifade edebileceği pek bir şey yoktur. Boyut sayısı arttıkça anlamsız görünen veriler anlam kazanmaya başlamaktadır. OLAP küpleri olarak adlandırılan bu çok boyutlu yapılar üzerinde dilimleme, döndürme, ayrıntıya inme gibi çok boyutlu analiz işlemleri yapılarak iş verisini işlemek mümkün olmaktadır.

3.3.1. Küp bileşenleri

3.3.1.1. Olay

Olay, ölçekler ve durum verisinden oluşan birbiriyle ilişkili veri parçaları yığındır. Her olay bir iş parçasını, iş hareketini veya iş sürecinde kullanılabilecek olan bir olayı temsil etmektedir. Bir şirketin satışları üç aylık dönem için mağaza bazında görülmek istendiğinde genellikle toplam gelir ve net gelire bakılmaktadır. Bunların her biri ilgili olaylardır [33].

3.3.1.2. Boyutlar

OLAP küplerinde en önemli kavram boyut kavramıdır. OLAP veri yapıları boyutsal olarak modellenmektedir. Kullanıcılar veriyi seçilen boyutlar açısından değerlendirmektedirler. Boyutlar bir organizasyondaki birbiriyle ilişkili faaliyetlerin belli bir ölçeğe göre sınıflandırılmasını sağlamaktadır [34,35].

Örneğin bir satış uygulaması için boyutlar şu şekilde olabilir [34]:

Zaman: Bu boyut hemen hemen tüm OLAP modellerinde yer almaktadır. Bu boyut organizasyonun: “Bu ay geçen aya göre satışlarımız ne kadar değişim göstermiştir?”, “Bu yıl yapılan harcamalar ile geçen yıl yapılan harcamalar arasında ne kadar fark olmuştur?” tipinde sorularına yanıt vermekte kilit rol oynamaktadır.

Ürün: Ürünler genellikle ürün tipine göre sınıflandırılmaktadır. Analist “Tüm gelirlerin ne kadarı Ürün A’dan elde edilmiştir? “ya da “En çok kazandıran ürün hangisidir?” tipinde soruların cevabını isteyebilir.

Lokasyon: Büyük organizasyonların farklı ülkelerde, bölgelerde, şehirlerde ve hatta her şehirde birkaç satış noktası olabilmektedir. Karar verici farklı bölgelerdeki satışların karşılaştırmasını yapmak isteyebilir.

Müşteri: Her organizasyon sahip olduğu müşteri yelpazesini anlamak ister. “En çok gelir getiren müşteri grubu hangisidir?”, “Müşterilerimizin çoğunluğunu hangi tip müşteriler oluşturmaktadır?” tipinde soruların cevabını bilmek bir organizasyon için önemlidir.

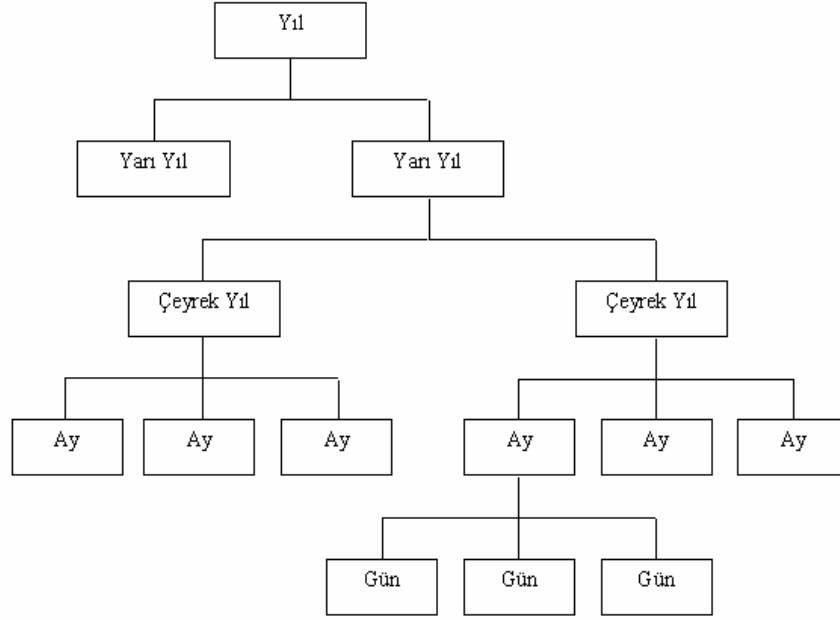
3.3.1.3. Boyut üyeleri

Bir boyut bir çok seviyeden oluşabilir. Zaman boyutu gün, hafta, ay, yıl seviyelerini içermektedir. Bu seviyelerden her biri üyelerden oluşmaktadır. Örneğin gün seviyesi Pazartesi, Salı, Çarşamba ya da Gün1, Gün2, Gün3...Gün7 şeklinde olabilir. Aynı şekilde aylar Ocak, Şubat, Mart v.b. üyelerden oluşabilmektedir. Bu şekilde bir boyut, boyut içerisinde seviyelerde düzenlenmiş birçok boyut üyesi içerebilmektedir [33].

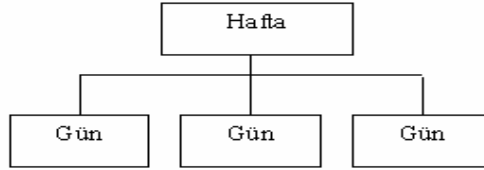
3.3.1.4. Boyut hiyerarşileri

Bir boyutun üyelerini bir ya da daha fazla hiyerarşi içerisinde sıralamak mümkündür. Aynı zamanda her hiyerarşi birden fazla hiyerarşi seviyesinden oluşabilmektedir. Bir boyut üyesinin birden daha fazla hiyerarşi yapısı içerisinde bulunabilmesi mümkündür [33].

Şekil 3.5.’te zaman boyutu için iki farklı hiyerarşi tanımlanmıştır. Veriye haftalık olarak kümelenecek şekilde bakılmak istendiğinde Şekil 3.5b’de görülen hiyerarşi düzeni kullanılmaktadır. İki farklı hiyerarşi tanımlanmasının sebebi bir haftanın iki farklı ay içerisinde yer alabilir olmasıdır.



(a)



(b)

Şekil 3.5. Zaman boyutunda farklı hiyerarşiler. (a) Aylık veriye göre hiyerarşi. (b) Haftalık veriye göre hiyerarşi.

Hiyerarşiler sayesinde analistin bir boyuttaki verileri önce özet seviyede inceleyip daha sonra farklı ayrıntı seviyelerine inmesi mümkündür. Örneğin ilk olarak tüm yıla ait satış verilerini inceleyip daha sonra üçer aylık zaman dilimi seviyelerine inerek veriyi inceleyebilmektedir [36].

3.3.1.5. Ölçek

Diğer bir önemli kavram ölçeklerdir. Ölçekler iş analiz konusunun niceliğini ifade etmektedir. Örnek olarak satış verileri verilecek olursa buradaki

önemli ölçekler kar, maliyet, birimler, iskonto v.b. olacaktır. Bu şekilde boyutlar içerisindeki karmaşık ham veri OLAP sayesinde tüm kullanıcılar tarafından anlaşılabilir olmaktadır. OLAP kar, maliyetler gibi ayrılmış ölçeklerin gerektiği karmaşık modellerin işlenmesine imkan tanımaktadır [34].

3.4. Boyutlu Modelleme Türleri

3.4.1. Yıldız (star) modeli

Veritabanı, merkezi bir olay tablosu ve bu merkezi tablonun etrafında düzenlenmiş boyut tablolarını içermektedir. Ortaya çıkan şekil yıldız biçimindedir. Bu modele yıldız modeli denilmesinin sebebi bu biçiminden dolayıdır. Böyle bir düzenlemede boyut tabloları merkezdeki olay tablosu ile doğrudan ilişkili olmaktadır [36].

Olay tablosu, satış miktarları gibi mevcut sayısal ölçüleri içermektedir. Boyut tabloları ise olay tablosunda var olan verilere işaretçiler içermektedir [37]. Boyutların kombinasyonu sonucu ortaya bir değer çıkmıyorsa bu boş değer yıldız modelinde depolanmamaktadır. Boyut tabloları herboyutu tanımlamakta ve her boyut olay tablosundaki veriyi bölümlendirmekte kullanılmaktadır [38].

Şekil 3.6'da yer boyut tablosu yer_anahtarı, sokak, şehir, ülke gibi niteliklerin bir kümesidir. Olay tablosunda yer alan her kayıt bir gün içerisinde bir bölgede satılan belirli bir ürünün toplam satışlarını göstermektedir. Ürün, yer veya günün farklı bir kombinasyonu ise olay tablosunda farklı bir kaydı oluşturmaktadır.

Zaman Boyut Tablosu

Zaman_anahtarı
Gün
Ay
Çeyrek
Yıl

Ürün Boyut Tablosu

Ürün_anahtarı
Ürün_ismi
Ürün_kodu
Ürün_kategorisi
Marka_ismi

Olay Tablosu

Zaman_anahtarı
Ürün_anahtarı
Yer_anahtarı
Satış_miktarı
Satış_tutarı

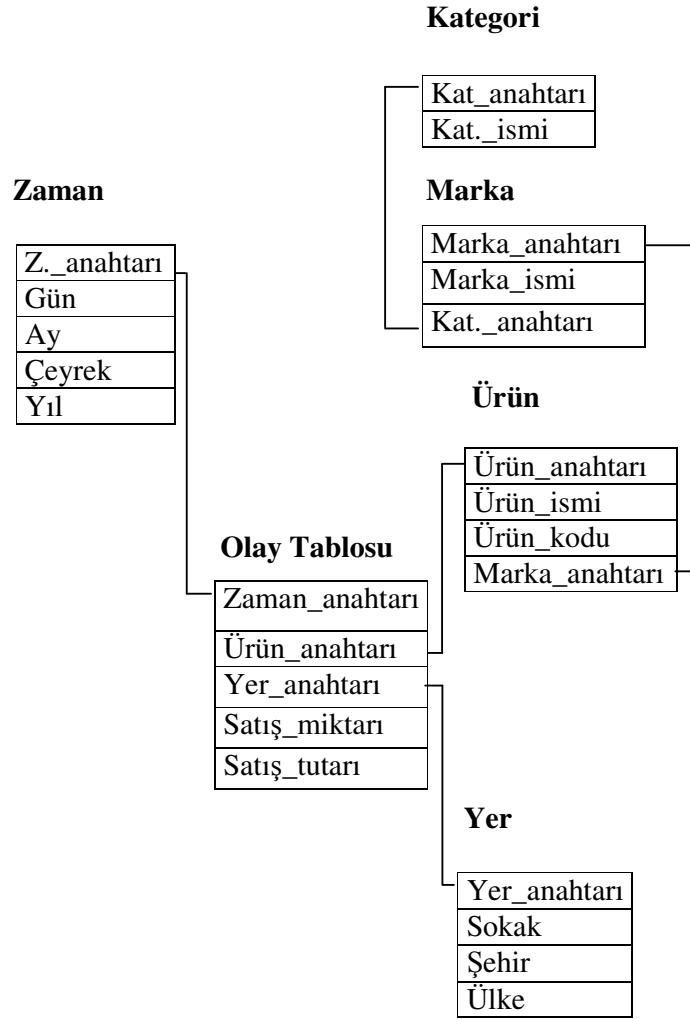
Yer Boyut Tablosu

Yer_anahtarı
Sokak
Şehir
Ülke

Şekil 3.6. Yıldız modeli

3.4.2. Kar tanesi (snowflake) modeli

Yıldız modelinin bir alternatifi olan kar tanesi modeli, veritabanı çok miktarda kategori içerdiğinde ve boyutlar büyük olduğunda tercih edilmektedir. Kar tanesi modeli boyut tablosunda yer alan nitelikler içinde ayrıca tablolar oluşturabilmektedir. Bu şekilde verilerin daha hızlı bir şekilde elde edilmesi sağlanmaktadır [37]. Şekil 3.7.'de kar tanesi modeli görülmektedir. Ürün boyutu üç ayrı tabloya ayrılarak marka ve kategori boyutları oluşturulmuştur.



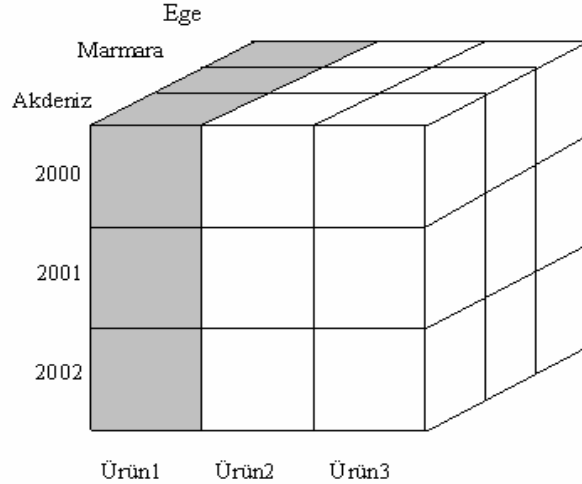
Tablo 3.7. Kar tanesi modeli

3.5. Temel OLAP İşlemleri

3.5.1. Slice ve dice işlemleri

Bir slice, boyutların bir ya da daha fazla üyesi için tek bir değerle ilgili olarak düzenlenmiş küpün bir altkümesidir. Slice işlemi tek bir boyutun seçimine ve küpün belli bir bölümü üzerinde odaklanılmasına dayanmaktadır. Örnek olarak zaman, bölge, ürün tipi veya bu boyutların herhangi bir kombinasyonunu vermek mümkündür [36,39].

Şekil 3.8.'de ürün boyutunun Ürün1 üyesi ile ilgili zaman ve bölge boyutunun tüm verileri seçilip slice işlemi yapılmıştır. Ürün1'in 2000, 2001 ve 2002 yıllarındaki Akdeniz, Marmara ve Ege Bölgesi'ndeki satış verilerini incelemek mümkündür. 2001 yılına ait tüm veriler incelenmek istendiğinde ise Şekil 3.9.'da olduğu gibi slice işlemi yapmak gerekmektedir.

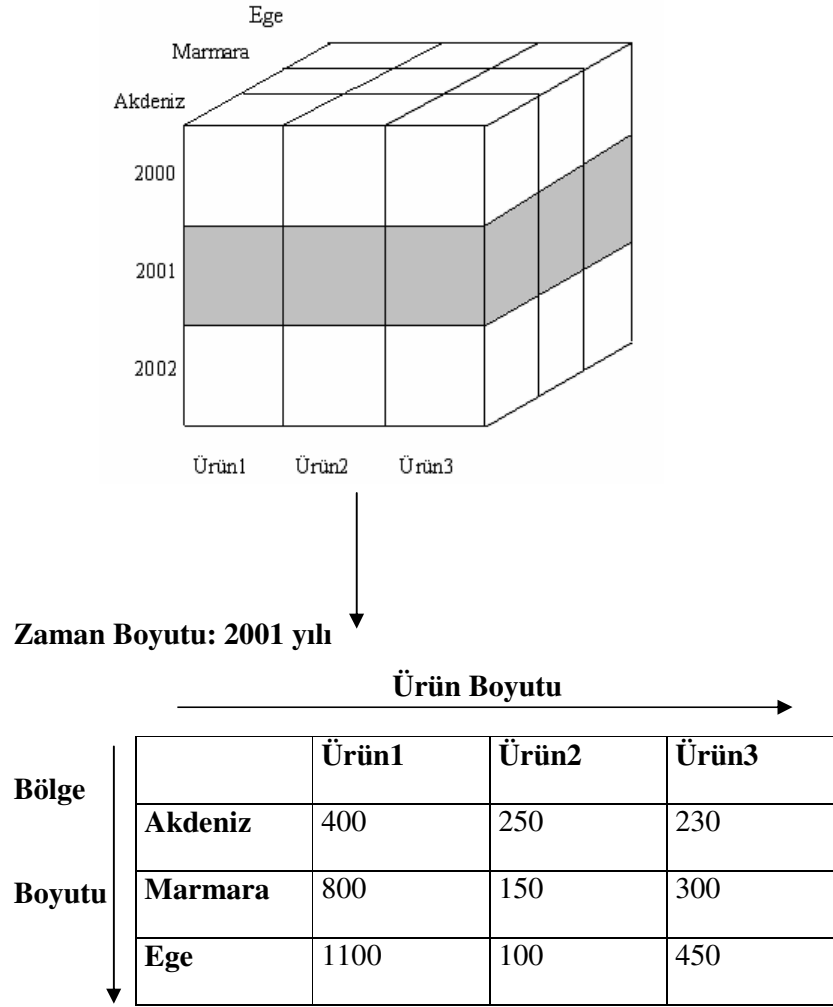


Ürün Boyutu: Ürün1

Bölge Boyutu

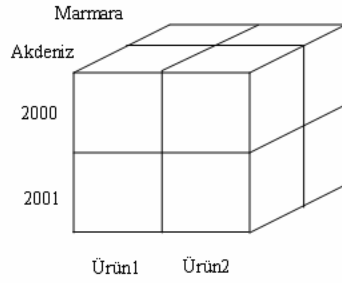
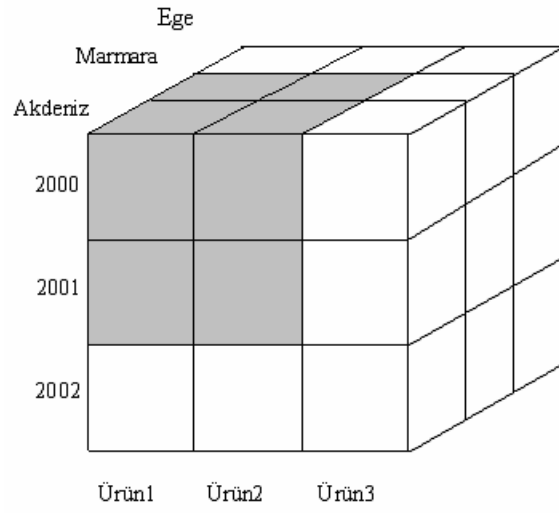
	Akdeniz	Marmara	Ege
2000	2750	3900	1850
2001	880	1250	1650
2002	3850	2150	1800

Şekil 3.8. Ürün1'e göre slice işlemi



Şekil 3.9. 2001 yılı verisi üzerinde slice işlemi

Dice işlemi ise iki veya daha fazla boyut üzerinde seçme işlemi gerçekleştirerek alt küpler oluşturulması işlemidir [28]. Şekil 3.10'da Ürün1 ve Ürün2'nin Akdeniz ve Marmara Bölgeleri'nde 2000 ve 2001 yıllarına ait verileri dice işlemi ile çekilerek bir alt küp oluşturulmuştur.



	Akdeniz		Marmara	
	Ürün1	Ürün2	Ürün1	Ürün2
2000	1300	900	3000	500
2001	400	250	800	150

Şekil 3.10. Dice işlemi ile alt küp oluşturma

3.5.2. Drill down ve roll up işlemleri

Drill down işlemi bir ya da daha fazla boyut hiyerarşisinde toplam seviyesinden detay seviyesine inilerek verilerin incelenmesi anlamına gelmektedir [40].

Ege Bölgesi'ne ait veriler 2001 yılının çeyreklerine göre incelenmek istendiğinde Şekil 3.11.'de olduğu gibi drill down işlemi ile detay seviyesine inilerek istenilen veriler elde edilmektedir.

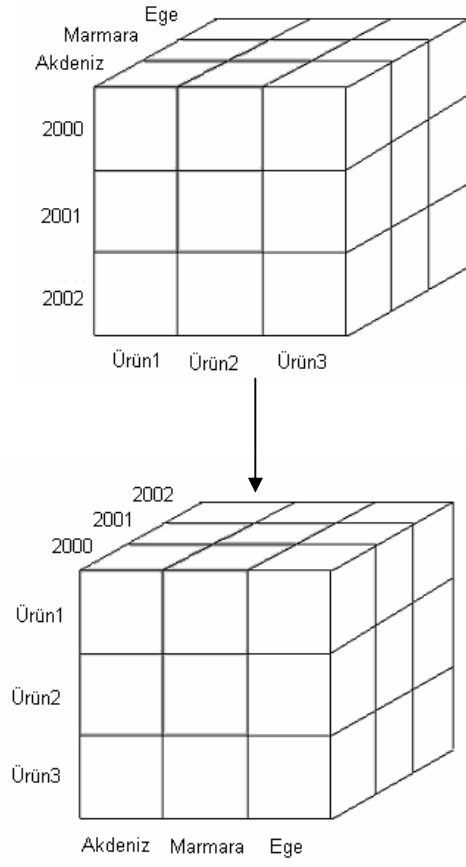
Roll Up işleminde ise küp verisi üzerinde kümeleme işlemi gerçekleştirilmektedir. Drill Down işleminin aksine detay seviyesinden toplam seviyesine çıkılarak veriler incelenmektedir [32].

Ege Bölgesi		Ürün 1	Ürün 2	Ürün3
2001 Yılı	1. Çeyrek	255	0	200
	2. Çeyrek	625	52	120
	3. Çeyrek	120	33	0
	4. Çeyrek	130	15	80

Şekil 3.11. 2001 yılı verisi üzerinde Drill Down işlemi

3.5.3. Pivot (Rotate) işlemi

Küp verisinin eksenleri etrafında döndürülerek veriye farklı açılardan bakılmasını sağlayan bir işlemdir [28]. Şekil 3.12a'da üç boyutlu veri küpü eksenleri etrafında döndürülerek verinin farklı bir görünümü sağlanmıştır. Benzer biçimde Şekil 3.12b'de olduğu gibi iki boyutlu veri yapıları üzerinde de pivot işlemi gerçekleştirilmek mümkündür.



(a)

	Ürün1	Ürün2	Ürün3
Akdeniz	400	250	230
Marmara	800	150	300
Ege	1100	100	450



	Akdeniz	Marmara	Ege
Ürün1	400	800	1100
Ürün2	250	150	100
Ürün3	230	300	450

(b)

Şekil 3.12. Pivot işlemi (a) Küp verisi üzerinde üç boyutlu pivot işlemi (b) İki boyutlu veri üzerinde pivot işlemi

3.5.4. Diğer OLAP işlemleri

Bazı OLAP sistemleri ek drilling işlemlerine izin vermektedir. Örnek olarak drill-across işlemi birden fazla olay tablosunu kapsayarak sorgulamalar yapılmasını sağlamaktadır. Drill-through işlemi ise veri küpünün en alt seviyesinde temel detay veriye yani ilişkisel tablolara inilmesine izin verir [28,33].

3.6. Codd Kuralları

İlk olarak 1993 yılında Dr. E. F. Codd tarafından kullanılan OLAP terimi gene Codd tarafından 12 temel madde ile tanımlanmıştır [13, 41]:

1. Çok boyutlu kavramsal görünüm: İş analistleri organizasyondaki olayları birçok değişkene bağlı olarak ve çok boyutlu şekilde kavramak isterler. Bu yüzden OLAP'ta kavramsal veri gösterimi slice, dice, ve pivot işlemleri gibi sezgisel operasyonlara izin verecek şekilde çok değişkenli ve çok boyutlu olmalıdır.

2. Açıklık: Kullanıcılar verinin depolanması için ne gibi kaynaklar kullanıldığını, veri işleme sürecini ve verinin nasıl düzenlendiğini bilmek zorunda değildirler. Ancak OLAP herkese açık ve anlaşılır bir yapıda olmalıdır. Kullanıcının analiz aracı ve sunucu arasındaki kesintisiz erişimine izin vermelidir.

3. Erişilebilirlik: Analistin ortak kavramsal şema dahilinde analiz yapma imkanının olması gerekir. Böylece veri olduğu şekliyle bulunduğu eski veritabanının kontrolünde kalmaya devam edecek, ortak analitik model değişmeyecek ve analist yalnızca analiz edilecek veriye erişecektir.

4. Tutarlı Raporlama Performansı: Analistler, veritabanı büyüklüğünün artmasıyla birlikte performans düşüklüğü ile karşı karşıya kalmamalıdır. Raporlama performansı, raporun tür ve içeriğine bağlı olmamalı, sabit olmalıdır.

5. İstemci-Sunucu mimarisi: Büyük miktardaki verilerin analitik işleme operasyonları için mainframe'lerde depolanması gerekmektedir, fakat bunlar aynı zamanda PC'lerde açılabilir. Böylece bir istemci, OLAP sayesinde istemci-sunucu ortamında gerçekleştirilebilir.

6. Genel boyutlandırılabilirlik: Tüm boyutlar sahip oldukları özellikler açısından birbirine denk olmalıdır. Ek işlev özellikleri seçilen boyutlara verilebilir.

7. Dinamik seyrek matris ayarlaması: OLAP aracı seyrek verilerin en iyi şekilde işlenmesini temin etmelidir. Erişim hızı veri hücrelerinin yerleşimine bağımlı olmadan korunabilmeli ve farklı sayıda boyutlara sahip modellerde sabit olmalıdır.

8. Çoklu kullanıcı desteği: Birden fazla analist bir analitik model üzerinde eş zamanlı olarak çalışmak veya aynı veri üzerinde çeşitli modeller oluşturmak zorunda kalabilmektedir. OLAP aracı bu erişimleri verebilmeli, bütünlüğü ve verinin korunmasını temin edebilmelidir.

9. Sınırlandırılmamış çapraz boyut operasyonları: Herhangi bir boyut üzerinde veri hesaplama ve düzenlenmesi engellenmemeli veya sınırlandırılmamalıdır.

10. Sezgisel veri düzenleme: Satır ve sütunlardaki verinin ayrıntılandırılması, birleştirme ve diğer veri düzenlemeleri uygun şekilde gerçekleştirilmeli ve kullanımı kolay bir arayüz olmalıdır.

11. Esnek raporlama: Değişik veri görüntüleme yöntemlerini desteklemeli. Diğer bir ifadeyle raporlar olası açılardan sunulabilmeli, istenilen boyutlar eklenebilmelidir.

12. Sınırsız boyutlar ve grup seviyeleri: Her OLAP aracı analitik modelde en az 15 boyut bulundurabilmelidir. Bu boyutlardan her biri kullanıcı tarafından tanımlanan sınırsız sayıda grup seviyesini kabul edebilmelidir.

3.7. OLAP Modelleri

Çok boyutlu analiz için esas verinin yüksek sorgulama performansını, ölçeklenebilirliği ve çoklu erişimi sağlayacak bir yapıda depolanmış olması gerekmektedir. İlişkisel veritabanları sık yapılan ve basit sorgulamalar için hazırlanmaktadır. Karmaşık ve çok boyutlu sorgulamalar yapmak için uygun değildir. Birçok sorgu tek bir SQL sorgusuyla yapılamamaktadır. Çok boyutlu sorgular daha fazla tablo birleşimini ve tüm tablonun incelenmesini gerektirmektedir. Tüm bunlar performansı büyük ölçüde düşürmektedir.

Çok boyutlu analiz yapılırken ilişkisel modelde var olan sınırlamalardan etkilenilmemesi için üç tip veri depolama stratejisi kullanılmaktadır:

- Özel çok boyutlu veritabanları (MDDDB) kullanılması, OLAP sorgularında verinin geri alınması ve depolanması için hazırlanmaktadır.

- Bir veri ambarı ilişkisel teknolojiden yararlanılarak yapılandırılır fakat operasyonel işlemler yerine karar destek için optimize edilir.

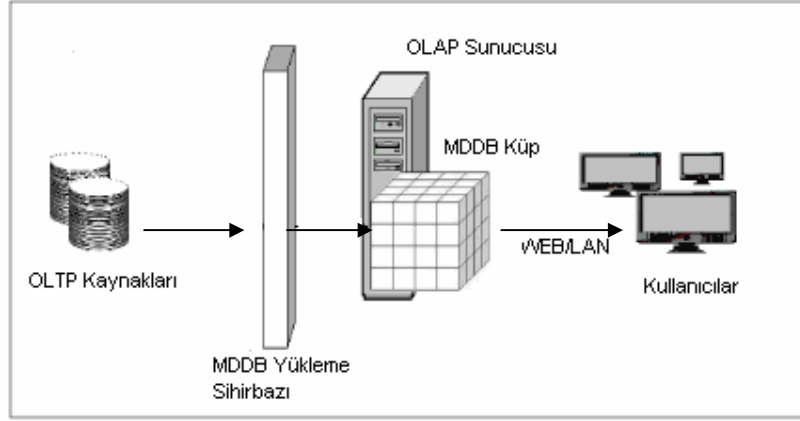
- İlk iki yaklaşımın birleşimi.

Çok boyutlu depolamayı sağlayan OLAP araçları çok boyutlu OLAP (MOLAP) olarak isimlendirilmektedir. İlişkisel veritabanlarında depolanmış veriye erişim sağlayan OLAP araçları ise ilişkisel OLAP (ROLAP) olarak isimlendirilmektedir. Bu iki yaklaşımın bir karışımı olan ve Karma OLAP (HOLAP) olarak bilinen OLAP araçları ise ilişkisel veritabanları ve çok boyutlu veritabanlarının önemli özelliklerinin birleştirilmesiyle elde edilmektedir [42].

3.7.1. Çok boyutlu OLAP (MOLAP)

MOLAP modelinde analiz edilecek veri çok boyutlu veritabanlarında depolanmaktadır. Bir MDDDB, operasyonel veritabanlarında bulunan büyük miktarda veri kullanılarak önceden hesaplanmış özet veriyi içermektedir. Önceden hesaplanmış özet veri OLAP için gereklidir, çünkü OLAP sorguları karmaşıktır ve doğrudan ham verinin bulunduğu operasyonel veritabanı üzerinden sorgulama yapmak saatler hatta günler alabilmektedir [14]. Çok boyutlu veritabanı yönetim sistemleri, veri ambarından çok boyutlu veritabanına veri yükleme işlemi boyunca özet küpler üretme ve birleştirme yeteneklerini sağlayan özel yazılım sistemleridir. Özet veriye ihtiyaç duyan kullanıcılar önceden birleştirilmiş veri sayesinde hızlı cevap almakta ve küp verisini sorgulamak için başka bir veritabanı kaynağına ihtiyaç duymamaktadırlar [31].

Çok boyutlu veritabanlarında çok boyutlu diziler kayıt yapılarını oluşturmaktadır. Örneğin ÜrünA için, 2001 yılının Ocak ayında, Mağaza1 mağazasında, Kanal05 dağıtım kanalı altında 500 birimlik ürün satışı verisi (ÜrünA, 2001/01, Mağaza1, Kanal05) değerleriyle temsil edilen bir dizide depolanmaktadır. Dizideki her hücre tüm boyutların kesişimi sonucu oluşturulmaktadır. Böyle bir tasarımda tüm hücreler bir değer içermeyebilir [36,38]. Şekil 3.13'te MOLAP mimarisi görülmektedir. Veri, kullanıcıların analizleri gerçekleştirebildiği bir MDDDB sunucusunda çok boyutlu küpler biçiminde depolanmaktadır [43].



Şekil 3.13. MOLAP Mimarisi

Önceden tanımlanmış boyutlardan oluşturulan veri küpleri içeren çok boyutlu veritabanları hızlı cevap alınmasını sağlamasına rağmen ilişkisel veritabanlarına göre hafıza, depolama ve işlemci gibi kaynakların daha fazla kullanımına neden olmaktadır [37].

3.7.1.1. Ön birleştirme

Çok boyutlu veritabanlarının en önemli avantajı çoğu sorgu için cevapları önceden birleştirebilmesidir. Oysa ilişkisel bir veritabanı “Geçtiğimiz çeyrekte kaç paket deterjan satışımız olmuştur?” gibi bir soruya cevap verebilmek için tüm veriyi toplamak ve tüm ilişkili kayıtları taramak zorundadır. Çok boyutlu bir veritabanı ise sadece çok boyutlu dizide ilişkili satır ve sütunlardaki hücreleri toplayarak böyle bir sorunun cevabını hızlı bir şekilde hesaplayabilmektedir. Hesaplama bir kere yapıldığında toplamlar dizi yapısında depolanabilmektedir.

Çok boyutlu veritabanları tüm veriyi önceden birleştirebilmektedir. Böylece her hiyerarşi seviyesi için toplamlar veritabanı yüklendiğinde hesaplanmaktadır. Bu yaklaşım çoğu sorgular için cevapların oldukça hızlı bir şekilde elde edilmesini sağlamaktadır. Fakat bu şekilde veritabanı için oldukça fazla yer ayrılması gerekmektedir ve veritabanının yüklenmesi daha uzun sürmektedir. Bu yüzden sadece sık olarak kullanılan toplamlar birleştirilmeli ve diğer hesaplamalar gerektiğinde yapılmalıdır. Hangi verinin önceden birleştirilmiş olması gerektiğine ve hangi verinin OLAP motoru tarafından hesaplanması gerektiğine karar verilmesi veritabanı performansı için önemlidir [42].

3.7.1.2. Detaylı bilginin alınması

Çok boyutlu veritabanlarının çoğu, ilişkisel veritabanlarında bulunan detay verinin istenildiği zaman elde edilmesi için veriye SQL yardımıyla erişime izin vermektedir. Bu bilgiler sadece görüntülenmek için kullanılabilir ve daha ileri analizler için kullanılamamaktadır [42].

3.7.1.3. Seyreklik (Sparsity)

Seyreklik hücrelerin bir değer içermemesi veya NULL değer alması anlamına gelmektedir. OLAP uygulamalarında girilen veri veya esas veri genellikle seyrek yapıdadır. Ayrıca yapıdaki boyut sayısı arttıkça yapı daha seyrek bir hale gelmeye başlamaktadır. Örnek verilecek olursa, ay ve ürüne göre satış verileri depolanmak istenildiğinde büyük olasılıkla her üründen her ay satış yapılacak ve veritabanının her hücresi veriyle dolu olacaktır. Ay ve ürün boyutuna ek olarak müşteri ve bölge boyutları eklendiğinde her müşteriye tüm ürünlerden tüm bölgelerde satış yapılamayacağı için hücrelerden bazılarında veri girişi yapılmayacak ve bu hücreler boş olacaktır [44].

Çok boyutlu veritabanının boş hücrelerle dolu olmamasına çalışılmalıdır. Bunu önlemek için çok boyutlu veritabanı üreticilerinin kendilerine özgü yöntemleri vardır. Genel olarak veritabanı NULL değerlerin depolanmasına gerek kalmayacak şekilde sıkıştırılmaktadır [42].

3.7.1.4. MOLAP modelinin avantaj ve dezavantajları

MOLAP modelinin avantaj ve dezavantajlarını şu şekilde özetlemek mümkündür [45].

Avantajları:

- Üstün performans: MOLAP küpleri verinin hızlı bir şekilde alımı için kurulmaktadır. Slice ve dice operasyonları için uygundur.
- Karmaşık hesaplamalar: Küp oluşturulduğunda tüm hesaplamalar önceden üretilmektedir. Bu nedenle karmaşık hesaplamalar hızlı bir şekilde geri alınabilmektedir.

Dezavantajları :

- Sınırlı miktarda veri kullanılabilir: Küp oluşturulduğunda tüm hesaplamalar üretilmiş olmaktadır. Küp içerisinde büyük miktarda veri içerilmesi mümkün değildir. Bu, küp verisinin büyük miktarda veriden alınamayacağı

anlamına gelmemektedir. Fakat bu durumda küp sadece özet seviyede bilgi içerecektir.

- Ek yatırımlar gerektirmektedir: Küp teknolojisi genellikle organizasyonda daha önceden mevcut değildir. Bu nedenle MOLAP teknolojisini adapte etmek insan ve sermaye kaynakları gerektirmektedir.

3.7.2. İlişkisel OLAP (ROLAP)

Geleneksel sorgu ve raporlama araçları genellikle tüm ham veriyi veri ambarı içerisine yüklemekte ve sorgu aracının sorgular yapmasına izin vermektedir. Sorgu sonuçlarının formülasyonu sorgu sırasında yapılmaktadır. Bu nedenle performans kabul edilemeyecek derecede düşük olmaktadır ve sonuçlar MDDB ya da ROLAP'ta elde edilenden daha basit seviyede olmaktadır. MDDB her sorgu sonucunu önceden hesaplayarak hızlı cevaplama zamanı sağlamaktadır ancak sistem yüklenmesinde ölçeklenebilirlik problemleri yaşanmaktadır. ROLAP sistemleri ön hesaplama ve çalışırken hesaplama yeteneklerinin her ikisine de izin vererek depolama ve performansı dengelemektedir [7].

ROLAP modelinde veri ilişkisel biçimde satır ve sütunlar halinde depolanmaktadır. Bu modelde veri kullanıcılara iş boyutları biçiminde sunulmaktadır. Kayıt yapısının kullanıcıya kapalı olması ve veriyi çok boyutlu sunmak amacıyla üstveri (metadata) anlamsal katmanı yaratılmaktadır. Üstveri katmanı boyutların tablolara eşleştirilmelerini sağlamaktadır. Bunlara ilave olarak üstveri özetlemeleri ve kümelemeleri desteklemektedir. Üstveriyi ilişkisel veritabanlarında depolamak mümkündür [36].

İlişkisel veritabanları çok boyutlu analiz için optimize edilmemelerine rağmen diğer yönlerden çok boyutlu veritabanlarına göre avantajlara sahiptirler. Özellikle büyük veri kümelerini ölçeklendirme ve yineleme, geri alma gibi destekler içermektedirler. [42].

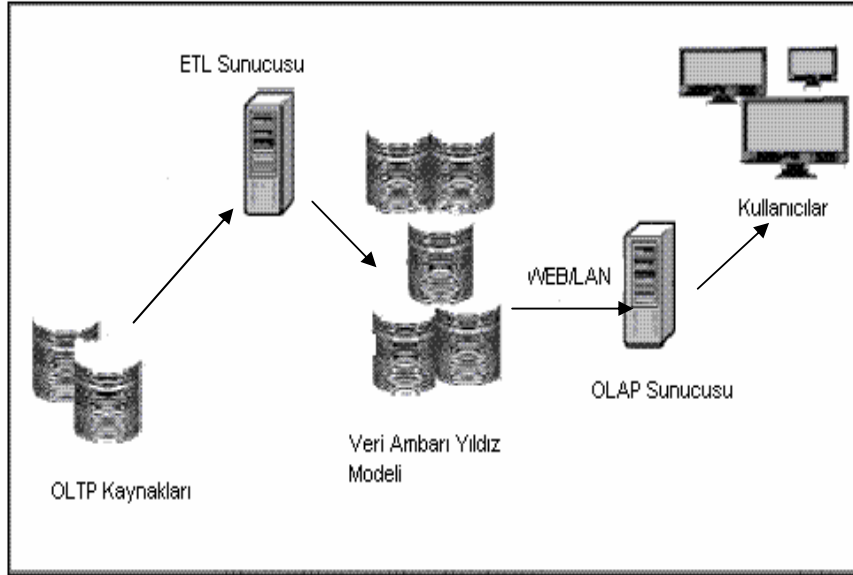
ROLAP ile şu özellikler sağlanabilmektedir [7]:

- Paralelizasyon: Büyük miktarda verinin depolanmasına izin vermekte ve karmaşık faaliyetler daha küçük bölümlere ayrılmaktadır. Bunların her biri paralel olarak yürütülebilmektedir.
- Veri bölümlenme: Verinin bir ya da daha fazla parçaya tablo ya da tablolara otomatik olarak dağıtılmasını sağlar, paralel operasyonlara izin vererek

ve büyük veri kümelerinin kullanımını kolaylaştırarak veritabanının yeteneklerini artırır. Aynı zamanda çoklu fiziksel depolama araçlarına verinin dağıtılmasına izin vermekte ve sistemin okuma/yazma performansını artırmaktadır.

Ölçeklenebilir, paralel, ilişkisel bir veritabanı depolama kapasitesini ve öncelikli veriye hızlı erişimi sağlamaktadır. Bu tür sistemler ilişkisel veritabanlarının performans, ölçeklenebilirlik ve açıklık gibi özelliklerini korurken analitik işlevselliğin avantajlarına da izin vermektedirler [7].

Şekil 3.14’de ROLAP mimarisi görülmektedir. Veri, ilişkisel bir veritabanında tablolar halinde tutulmaktadır. Tasarım yıldız modeli ya da kar tanesi modeli kullanılarak yapılandırılmış olabilir. ETL (extract, transform, load) sunucusu operasyonel sistemlerden gelen ham veri üzerinde seçme ve dönüştürme işlemlerini gerçekleştirerek veri ambarına uygun hale getirmekte ve yüklemektedir [43].



Şekil 3.14. ROLAP Mimarisi

3.7.2.1. ROLAP modelinin avantaj ve dezavantajları

ROLAP modelinin avantaj ve dezavantajlarını sıralamak gerekirse [45],

Avantajları :

- Büyük miktarlarda veri kullanılabilir: ROLAP teknolojisinde veri büyüklüğü sınırı, temel ilişkisel veritabanının veri büyüklüğü ile sınırlıdır. Diğer bir deyişle ROLAP veri miktarı üzerinde herhangi bir sınırlama belirlememektedir.

- İlişkisel veritabanlarında mevcut olan fonksiyonların kullanılabilirliği: Çoğunlukla ilişkisel veritabanı birçok fonksiyonla birlikte gelmektedir. ROLAP teknolojisi ilişkisel veritabanı üzerine kurulduğundan ilişkisel veritabanında var olan fonksiyonlara sahip olmaktadır.

Dezavantajları :

- Performans yavaş olabilir: Her ROLAP raporu gerçekte ilişkisel veritabanındaki bir SQL sorgusudur (ya da çok geçişli SQL sorgusu). Temel veri miktarı fazlaysa sorgu zamanı uzun olabilmektedir.

- SQL tarafından sınırlandırılmaktadır: ROLAP teknolojisi ilişkisel veritabanını sorgulamak için SQL ifadeleri üretmektedir ve bu SQL ifadeleri tüm ihtiyaçları karşılayamamaktadır SQL kullanarak karmaşık hesaplamalar gerçekleştirmek zordur. Bu şekilde ROLAP teknolojisi SQL'in yapabilecekleri ile sınırlı kalmaktadır. ROLAP üreticileri kullanıcıların kendi fonksiyonlarını tanımlamalarına izin veren yetenekler gibi özellikler ekleyerek bu dezavantajları azaltmaktadırlar.

3.7.3. MOLAP ve ROLAP modellerinin karşılaştırılması

MDDDB tipik olarak veriye doğrudan erişimi sağlayan indeksler içermektedir. Boyutsal bir problem çözülmeye çalışıldığında daha hızlı yanıtlar alınmaktadır. Buna rağmen hafızanın işgal edilmesi, veri sıkıştırma teknikleri kullanılması ve seyrek veriden dolayı önemli performans farklılıkları yaratmaktadır. ROLAP modelinde ise sorgulara daha yavaş yanıt alınmaktadır [7].

Çok boyutlu veritabanı sunucuları 50 GB büyüklüğünde depolamaya izin verirken RDBMS sunucuları terabyte büyüklüğünde depolamaya izin vermektedir[7].

Çok fazla seyrek veri söz konusu olduğunda ROLAP modelinin seçilmesi daha uygun olacaktır. MDDB’da veri güncelleştirilmesi yapıldığında özet veri otomatik olarak hesaplamaları yaparak güncelleştirmeleri yapmaktadır. Tablo 3.2. de MOLAP ve ROLAP modellerinin karşılaştırılması yer almaktadır.

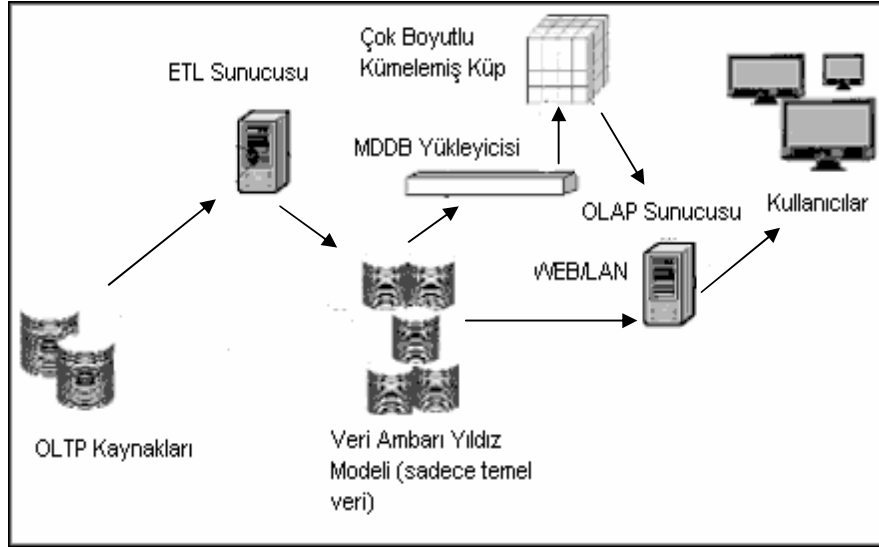
Tablo 3.2. MOLAP ve ROLAP modellerinin karşılaştırılması

Özellik	MOLAP	ROLAP
Hesaplama gücü, karmaşıklık	✓	
Seyrek veri		✓
Veritabanı güncellemesi	✓	
Veri değişkenliği		✓
Veri hacmi		✓
Geliştirme zamanı		✓
Sorgu yanıt zamanı	✓	
Tutarlılık, güvenilirlik	✓	
Veri yükleme zamanı		✓
Güvenlik		✓

3.7.4. Karma OLAP (HOLAP)

HOLAP, çok boyutlu bir veritabanı ve ilişkisel bir veritabanında depolanmış verinin eşzamanlı olarak çok boyutlu analizine imkan veren bir üründür. ROLAP ve MOLAP çözümlerinin önemli özelliklerini birleştiren HOLAP yaklaşımı, yerel ve uzak sunucularda bulunan farklı veri kaynaklarına erişim imkanı sunmaktadır [46].

Şekil 3.15’de HOLAP mimarisi gösterilmektedir. ROLAP ve MOLAP modellerinin önemli özelliklerini birleştirerek kullandığından dolayı bu modele HOLAP (Karma Model) ismi verilmektedir. Diğer bir deyişle bir küp için gerekli öncelikli veri ilişkisel bir veritabanında depolanmaktadır. Kümelenmiş veri ise yüksek performanslı çok boyutlu bir yapıda depolanmaktadır.



Şekil 3.15. HOLAP Mimarisi

HOLAP modeli kullanılarak yapılan depolama, temeli oluşturan ayrıntılı verinin kopyalanmasına gerek kalmaksızın kümelemeler yapmak için MOLAP modelinin avantajlarını sunmaktadır [25].

HOLAP, MDDB’de bulunan veri üzerinde çok boyutlu analizler yapılmasına izin vermektedir. Bununla birlikte kullanıcılar detay veriye ulaşmak istediklerinde HOLAP motoru, RDBMS kaynağından verinin alınarak son kullanıcılara iletilmesi için SQL sorguları üretmektedir [47].

Sorgulama performansı ROLAP modelinde olduğundan daha yüksektir fakat MOLAP modelindeki performans seviyesine ulaşamamaktadır. Detay veriye yapılan sorgular ilişkisel veritabanı sistemi kaynağına geçirilmektedir. HOLAP küpleri hemen hemen MOLAP küplerinde olduğu kadar işleme zamanı gerektirmektedir [25].

3.7.5. Masaüstü OLAP (DOLAP)

Bu yöntemde çok boyutlu mikro küpler oluşturulmakta ve kullanıcının makinesine yüklenmektedir. MOLAP’ın aksine büyük ölçekli bir küp oluşturmaya

gerek yoktur. Bunun yerine kullanıcı sorguyu gerçekleştirdiği anda dinamik olarak küp oluşturulmakta ve yerel bir küp içerisinde saklanmaktadır. MOLAP ve ROLAP ile karşılaştırıldığında küp içerisindeki veri hacmi ve çok boyutlu hesaplamalar sınırlıdır [43, 48].

4. KARAR DESTEK SİSTEMLERİNDE WEB TEKNOLOJİLERİ

Web tabanlı KDS, karar destek bilgisini veya karar destek araçlarını yöneticilere, iş analistlerine veya müşterilere Netscape Navigator veya Microsoft Internet Explorer benzeri web tarayıcılar kullanarak dağıtan bilgisayarlı sistemlerdir [4].

Web tabanlı KDS'leri, geniş bir coğrafyaya yayılmış olan organizasyonlarda teknolojik engelleri azaltarak kararların zamanında alınmasını sağlamaktadır. Web, ağ yapısından ve oldukça düşük yazılım, kurulum ve bakım maliyeti gerektirmesinden dolayı birçok organizasyonun istemci-sunucu platformu haline gelmektedir. Bu nedenle günümüzde organizasyonların büyük bir çoğunluğu zamanla ilgili, coğrafi ve teknolojik engellerin üstesinden gelmek amacıyla bir tarayıcı, web sunucusu ve veritabanı içeren çok katmanlı mimariye sahip karar destek uygulamaları kurmaktadır. Karar destek yazılımı üreticileri ürünlerine web teknolojilerini entegre etmektedirler [37, 49].

Organizasyonların web teknolojisini üç şekilde kullanabilmeleri mümkündür:

- İnternet: Kişisel veya organizasyon çapında kullanıcıların bilgi iletişim servislerine bir sınırlama olmadan erişimine izin veren herkese açık ve küresel bilgisayar ağı sistemidir. Organizasyondaki veriyi elde etmek isteyen herhangi bir kullanıcının organizasyonun Universal Resource Locator (URL)'si aracılığıyla istediği veriye ulaşabilmesi mümkün olmaktadır [37, 50].

- Intranet: Intranet, internet teknolojilerini kullanmaktadır. İnternette farklı olarak intranet herkesin erişimine izin vermemektedir. Sadece organizasyon içindeki kullanıcıların erişimine açıktır. Intranet kullanımının en önemli avantajı gizliliğin ve bilgi güvenliğinin sağlanmasıdır. Bir güvenlik duvarı tarafından izinsiz kullanıcıların erişimi engellenmektedir [37] .

- Extranet: Bir organizasyon, sahip olduğu intranet yapısını seçtiği iş ortaklarına veya organizasyonlara açık hale getirdiğinde bu yapı extranet haline gelmektedir. Extranet, internet ve intranet arasında bir yapıdır. Sahip olunan veriye farklı organizasyonların erişimine izin vermektedir [37, 50].

Web teknolojileri sayesinde herhangi bir yerdeki kullanıcının istediği veriye ulaşabilmesi mümkün olmaktadır. Organizasyon içindeki yöneticiler

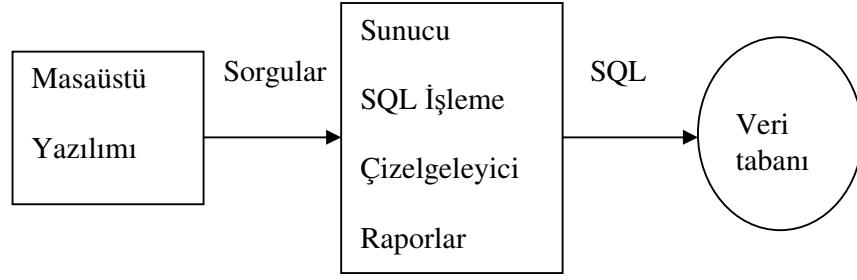
intranet üzerinden, organizasyon dışındaki tedarikçiler extranet üzerinden veya dünyanın herhangi bir bölgesindeki müşteriler internet üzerinden istedikleri verilere ulaşabilmektedirler [4].

İntranet ve extranet'in kullanıcılara ve organizasyonlara getirdiği avantajları şu şekilde sıralamak mümkündür [36]:

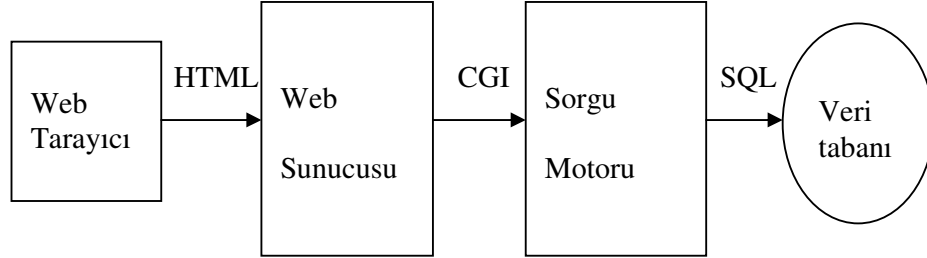
- Bilgiye erişim için ek bir çalışmaya gerek yoktur. Bir tarayıcı ile ulaşılabilir olmaktadır.
- Tarayıcı herhangi bir sistemle çalışabilmektedir.
- Web teknolojileri çoğu bilgi formatını kullanıcılara sunar. Metin, resim, grafik ve ses dosyalarının alınmasına izin verir.
- Organizasyonun, sahip olduğu veri ambarını extranet kullanarak iş ortaklarına açmasına izin verir.
- Veri ambarlarını web temelli hale getirmek için gerekli kurulum ve bakım maliyetleri düşüktür.

4.1. Web İşlem Modeli

Web mimarisi iki katmanlı ya da üç katmanlı istemci sunucu mimarilerden daha karmaşık bir mimariye sahiptir. Web işlemlerini yerine getirmek için ilave katmanlar gerekebilir. En azından bir web tarayıcı, veritabanı ve tarayıcı ile veritabanı arasındaki iletişimi sağlayan bir web sunucusu gerekmektedir [36]. Şekil 4.1. de geleneksel yöntem ve web işlem modelleri görülmektedir.



(a)



(b)

Şekil 4.1. Geleneksel ve web işlem modelleri (a) Üç katmanlı istemci/sunucu modeli (b) Web işlem modeli [37]

Şekil 4.1b 'deki modelde web tarayıcısından gönderilen HTML sayfaları HyperText Transfer Protocol (HTTP) kullanılarak web sunucusuna iletilmektedir. CGI script'leri kullanan web sunucusu HTML sayfaları ile gelen istekleri sorgu motoruna iletir. Sorgu motoru ise bu istekleri SQL sorguları haline çevirmektedir [37].

4.2. Karar Destek Sistemlerinde Kullanılan Web Araçları

Web temelli KDS'leri oluşturmak için kullanılan araçlara her geçen gün yenileri eklenmekte ve giderek karmaşık bir yapıya sahip olmaktadır. Bu araçlardan en önemlileri olarak HyperText Markup Language (HTML), Extensible Markup Language (XML), Active Server Pages (ASP), Common

Gateway Interface (CGI), Java applet'leri, Javascript ve ActiveX benzeri araçları vermek mümkündür.

HTML, bir web belgesinin mantıksal düzenlemesini belirlemek amacıyla tasarlanan ve bir nesneden diğer bir nesneye bağlantılar sağlayan komutlar içeren bir programlama dili yorumlayıcısıdır. Web belgeleri girdinin alınması ve çıktının karar destek programında görüntülenmesi amacıyla kullanılmaktadır [4].

XML, bilgisayarlar üzerinde depolanan, XML belgeleri olarak isimlendirilen veri nesnelere sınıfını ve bu nesnelere işleyen programların hareketlerini belirtmektedir. Veritabanları ve web belgeleri içeren KDS'lerinin büyük bir çoğunluğunda kullanılabilir. HTML'e benzemesine rağmen XML etiketleri her web sayfasını daha yapılandırılmış belgelere dönüştürmektedir. Örnek vermek gerekirse bir organizasyonda personel bilgilerini içeren bir belge içerisinde <isim> </isim>, <maaş> </maaş> benzeri etiketler kullanmak mümkündür. XML, KDS'nin belgeleri, veriyi ve bilgiyi daha hızlı ve etkili işlemesine izin vermektedir [4,51].

ASP kullanılan tarayıcının türünden bağımsız olarak sunucu tarafından yürütülen, dinamik ve etkileşimli web sayfaları hazırlanmasına izin veren bir teknolojidir [52].

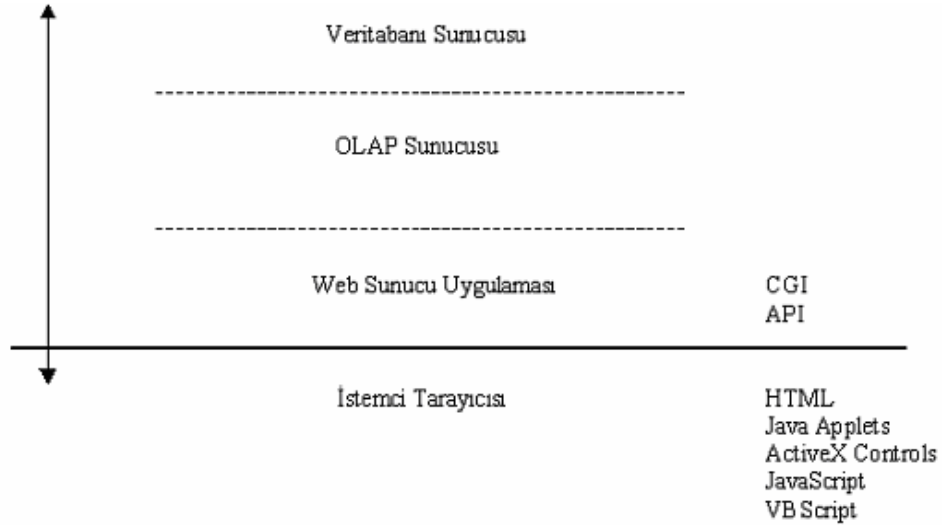
CGI uygulamaları, web sunucuları ve diğer uygulamalar arasında etkileşim sağlamak amacıyla oluşturulan ve sunucu tarafından yürütülen programlardır. Web formlarından değerleri almak ve veritabanları gibi uygulamalara web temelli bir arabirim sağlamak için kullanılmaktadır [4].

Java Applet'leri web tabanlı ortamlarda kullanmak amacıyla Sun Microsystems tarafından geliştirilen Java dilinde yazılmış, HTML belgeleri içerisinde kullanılabilen ve etkileşimli bir ortam sağlayan küçük programlardır. JavaScript'ler Java Applet'lerinde olduğu gibi Java dilinde yazılan komutlardan oluşmaktadır. ActiveX Controls Java Applet'lerine benzemektedir. Farklı uygulamalar arasında bilgi paylaşımını sağlamak amacıyla Microsoft tarafından geliştirilmiştir [4,30].

4.3. Web OLAP (WOLAP) Araçları

Organizasyonlarda giderek artan eğilim hiyerarşik yapının azaltılması ve gücün dağıtılması yönündedir. Amaç organizasyonun her kademesinin karar verme sürecinden haberdar edilmesi ve kararların hızlı bir şekilde alınabilmesini sağlamaktır. İş zekası araçları, web OLAP araçları (WOLAP) sayesinde bu araçlara erişmek ve kullanmak isteyen herkes tarafından bir web tarayıcı yardımıyla kullanılabilir olmakta ve organizasyonun her kademesinin karar verme sürecine dahil edilmesi sağlanmaktadır [53].

Büyük ve karmaşık uygulamalarda az sayıda kullanıcı olduğunda geleneksel istemci-sunucu mimarisini kullanmak sorun yaratmamaktadır. Kullanıcı sayısı çoğaldığında ise uygulamayı çok sayıda makineye yerleştirmek sorun olmaktadır. OLAP türü araçlar çok sayıda kullanıcı tarafından kullanıldığında web temelli uygulamalar ile bu sorun çözülebilmektedir. Bir web temelli uygulama istemci makinesinde kurulum gerektirmemektedir. Gerekli olan herşey bir web tarayıcısı ve intranet veya internete bir ağ bağlantısıyla sağlanabilmektedir [54]. Şekil 4.2’de tipik OLAP mimarisi görülmektedir. Web sunucusu genellikle CGI script’leri, web sunucu API (Application Program Interface)’leri, uygulama API’leri ve veritabanı API’lerinin ve benzeri bileşenlerin



Şekil 4.2. Tipik Web OLAP mimarisi [55]

bir kombinasyonu aracılığıyla tarayıcı ve OLAP uygulaması veya veritabanı sunucusu arasındaki iletişimi yönetmektedir. İstemci tarafı ise HTML, Java Apletleri, ActiveX, JavaScript ve VBScript benzeri bileşenlerin kombinasyonlarından oluşan fonksiyonlara ve bir arayüze sahiptir. OLAP, bir uygulama sunucusu olarak ağ üzerinde kurularak bu bileşenlerle birlikte çalışmaktadır.

Web ortamında OLAP araçlarının kullanılması yukarıdan aşağı tasarım yaklaşımını gerektirmektedir. Üreticiler ilk olarak kullanıcıların bir tarayıcıda rapor bilgisini hangi yollarla görüntüleyeceklerine karar verirler. Daha sonra OLAP motoruna tarayıcı temelli erişimi sağlamak için gerekli işlemler ayrıntılarıyla planlanır. Bu işlemler bilginin yayımlanmasını, veri ve işlevsellik için veritabanı veya uygulama sunucusuna HTTP erişiminin sağlanmasını ve analitik işlevsellik sağlayacak arayüzlerin tasarlanmasını içermektedir.

Yayın tarayıcıda görüntülenecek bilgiyi hazırlamak ya da oluşturmak için gerekli olan adımları içermektedir. Yöneticiler sunulacak rapor ve grafikleri yayım araçları kullanarak oluşturmaktadırlar. Bu araçlardan bazıları programcı araçları olarak yerleştirilmektedir. Örnek olarak HTML etiketleri web sunucusuna veritabanından hangi veriyi çekmesi gerektiğini belirtebilir. Benzer şekilde programlar tarayıcıya gönderilecek bilginin yapılandırılması için gerekli HTML fonksiyonlarını sağlayacak araç seti kullanabilirler. Diğer yayım araçları varolan ürünlere ilave edilebilmektedir. Raporlar oluşturmak isteyen yöneticiler için varolan bu ürünlere “rapor olarak kaydet” seçeneği eklenmiş olabilir [55].

4.3.1. Web OLAP yöntemleri

Web üzerinden OLAP işlevselliğine erişim sağlayan yöntemleri üç kategoriye ayırmak mümkündür [36,55]:

1. Önceden oluşturulan HTML belgeleri
2. Tarayıcıyı ActiveX veya Java bileşenleri ile desteleyerek
3. Sunucuda OLAP

4.3.1.1. Önceden oluşturulan HTML belgeleri

Önceden hesaplanmış ve biçimlendirilmiş HTML benzeri belgeler kullanılmaktadır. Bu belgeler analitik operasyonların sonuç setlerini içermektedir. İşlemler navigasyon araçları ile statik olarak yapılmaktadır. Bu yöntemde statik raporların web tarayıcıya taşınarak görüntülenmesi hızlı bir şekilde olmakla birlikte kullanıcının veriyle gerçek bir etkileşimi söz konusu değildir [36].

4.3.1.2. Tarayıcıyı ActiveX veya Java bileşenleri ile desteleyerek

ActiveX kontrolleri ve Java Applet'leri güçlü bir erişim yöntemi sağlamaktadır. Bu bileşenler istemci makinesinde çalışırlar. Veri işleme ve sunumunu istemci makinesinde gerçekleştirdiklerinden sunucu ve istemci arasındaki iletişimi en aza indirmektedirler [56].

Bu yöntemde OLAP işlevselliği önceden oluşturulan belgelerle hazırlanan raporlamaya göre daha fazla sağlanmaktadır.

4.3.1.3. Sunucuda OLAP

En iyi yöntem tüm OLAP işlemlerinin bir sunucu üzerinde yapılmasıdır. Arayüz bileşeni sunucu ile doğrudan iletişindedir. Kullanıcı istekte bulunduğu anda etkileşimli olarak veri istemciye iletilmektedir. İstemci sonuçları sunucudan aldığı anda bunları ayrıştırarak görüntülemek için bir nesne içine taşımaktadır [55].

Bu yöntemde istemci makinesinden bağımsız olarak bütünleşik sunucu ortamı sağlanmaktadır. Uygulamalar ve mantık sunucu üzerinde bulunduğu için fazla bakım gerektirmemektedir [36].

4.3.2. Web OLAP yaklaşımının yararları

OLAP analiz uygulamalarını web yoluyla kurumsal ve küresel çapta yürütmenin işletme açısından birçok avantajı bulunmaktadır. En önemli avantajlarından biri olarak yazılımın merkezi olarak kurulmasından dolayı daha az bakım gerektirmesini verebilmek mümkündür. Çoğu organizasyonlar yüzlerce birimde binlerce yazılım kopyasının bakımını yapmak zorunda kalmaktadır. Organizasyonlar web tarayıcılarının kullanılmasıyla bu yükten büyük ölçüde kurtulmaktadır. Web OLAP yaklaşımı organizasyonlara büyük ölçüde zaman kazandırmakta, yönetimi rahatlatmakta ve maliyetleri düşürmektedir.

OLAP uygulamalarını web temelli olarak kurmanın diğer bir avantajı ise organizasyon çapında bir standart yerleştirmektir. Analizler her birimde yaygın

olarak kullanılan tarayıcılar tarafından gerçekleştirilmekte, çalışanlara bu konuda eğitim vermek kolaylaşmakta ve böylece kullanıcılar konsolide edilmiş OLAP araçları ve uygulamalarını rahatlıkla kullanabilmektedir. Sonuç olarak elde edilen raporların kalitesinde bir artış sağlanmakta ve son kullanıcıların eğitim maliyetleri düşmektedir [57].

5. UYGULAMA

Uygulamanın amacı OLAP küplerinden oluşturulan analizlerin web tabanlı ortamda kullanıcılara sunulmasıdır. OLAP küplerinin oluşturulmasında SQL Server 2000 Analiz Servisi kullanılmaktadır. Yapılan analizlerin web tabanlı ortamda kullanıcılar arasında paylaşılması amacıyla Cubularity Knowledge Platform istemci uygulaması kullanılmıştır.

5.1. Bir OLAP Uygulaması Olarak SQL Server 2000 Analiz Servisleri

1990'ların başlarında OLAP araçları için en büyük problemlerden biri son kullanıcıların kullanımı için yeterince kolay anlaşılır olmaması idi. Son kullanıcılar genellikle veri ambarında toplanmış olan tarihsel verinin analizine dayanarak kararlar veren kişidir. Eskiden geliştiriciler son kullanıcıların bu araçları kısa sürede öğrenebilmesi ve kolay kullanabilmesi için pahalı uygulamalar kurmak zorundaydılar.

Microsoft firması veri ambarlamada kullanılmak üzere daha iyi ve özel tasarlanmış bir veritabanı sisteminin ve veri analizinin kolay bir şekilde yapılabilmesini sağlayacak araçların gerekliliğinin farkına vardı. Microsoft SQL Server 7.0 ve onun getirdiği araçlarla birlikte probleme geçerli, uygulanabilir bir çözüm getirilmiş oldu. Microsoft, SQL Server 7.0 ürününde OLAP Servisleri olarak bilinen araçlara, SQL Server 2000 ile birlikte yeni versiyon OLAP servislerini ve veri madenciliği yeteneklerini de ekleyerek bunları Analiz Servisleri olarak adlandırmıştır.

5.1.1. Microsoft Analiz Servisi'nin özellikleri

Microsoft SQL Server 2000 Analiz Servisi, OLAP ve veri madenciliği için bir orta katman sunucusunu içermektedir. Analiz Servis sistemi, küp bilgisine hızlı istemci erişiminin sağlanması ve analiz için çok boyutlu küp verilerinin yönetilmesinden sorumlu bir sunucu içermektedir. Analiz Servisi, karmaşık analitik sorgulara hızlı cevap verilmesini sağlamak amacıyla veri ambarında bulunan veriyi önceden hesaplayıp kümeleyerek küp içerisinde düzenlemektedir. Aynı zamanda Analiz Servisi, hem OLAP hem de ilişkisel veri kaynaklarından veri madenciliği modelleri oluşturulmasına izin vermektedir.

Analiz servisinin özelliklerini şu şekilde sıralamak mümkündür:

- Kolay kullanım: Birçok sihirbaz, editör ve yardımcı araçlar her aşamada kullanıcıların kullanımına sunulmuştur. Analysis Manager olarak bilinen kullanıcı arayüzü sadece meta veri görünümüleri ve küp verisine erişim olanağı sağlamakla kalmamakta aynı zamanda küpler, boyutlar ve seviyelerin düzenlenmesi ve oluşturulması için birçok sihirbaz içermektedir.
- Esneklik: Microsoft OLAP sunucusu sayesinde birçok depolama biçimi uygulanabilmektedir. Küp verisi çok boyutlu küp dosyalarına (MOLAP), ilişkisel veritabanlarına (ROLAP) veya bu iki yöntemin bir karışımı şeklinde depolanabilmektedir. Aynı zamanda küp bölümlendirilebilmekte ve her bölüm ayrı depolama yöntemiyle depolanabilmektedir.
- Ölçeklenebilirlik: OLAP istemcisi Windows 9x, Windows NT ve Windows 2000 işletim sistemlerinde çalışabilmektedir. Aynı zamanda Analiz Servisi kümeleme seçeneklerinin özelleştirilmesi, kullanım temelli optimizasyon, veri sıkıştırma ve dağıtık hesaplamalar gibi çeşitli veri ambarlama senaryolarına yer vermek amacıyla tasarlanmıştır. Tüm bu özellikler Microsoft Analiz Servisi'ni yüksek düzeyde ölçeklenebilir hale getirmektedir.
- Entegrasyon: Analiz Servisi, MMC ile bütünleşik haldedir. Analysis Manager MMC konsoluna bir snap-in olarak eklenebilmektedir. Analiz servisinde güvenlik SQL Server ve Windows NT güvenliği ile bütünleşik haldedir. Bu entegrasyon OLAP servisleri için OLE DB sağlayıcısına sahip olunması ve istemci ve sunucu taraflı ön belleklere sahip olunmasının bir sonucu olarak OLE DB veri kaynaklarının güçlü bir şekilde kullanımına izin vermektedir. Birçok üçüncü parti yazılım üreticileri Microsoft Analiz Servisi ve ürünleri arasında entegrasyon sağlamaktadır.
- Geniş çapta desteklenen API'ler ve fonksiyonlar: OLAP sunucusu ve Microsoft PTS, OLE DB, ADO MD, kullanıcı tanımlı fonksiyonlar ve karar destek nesnelerini desteklemektedir.

5.1.2. Analiz Servisi mimarisi

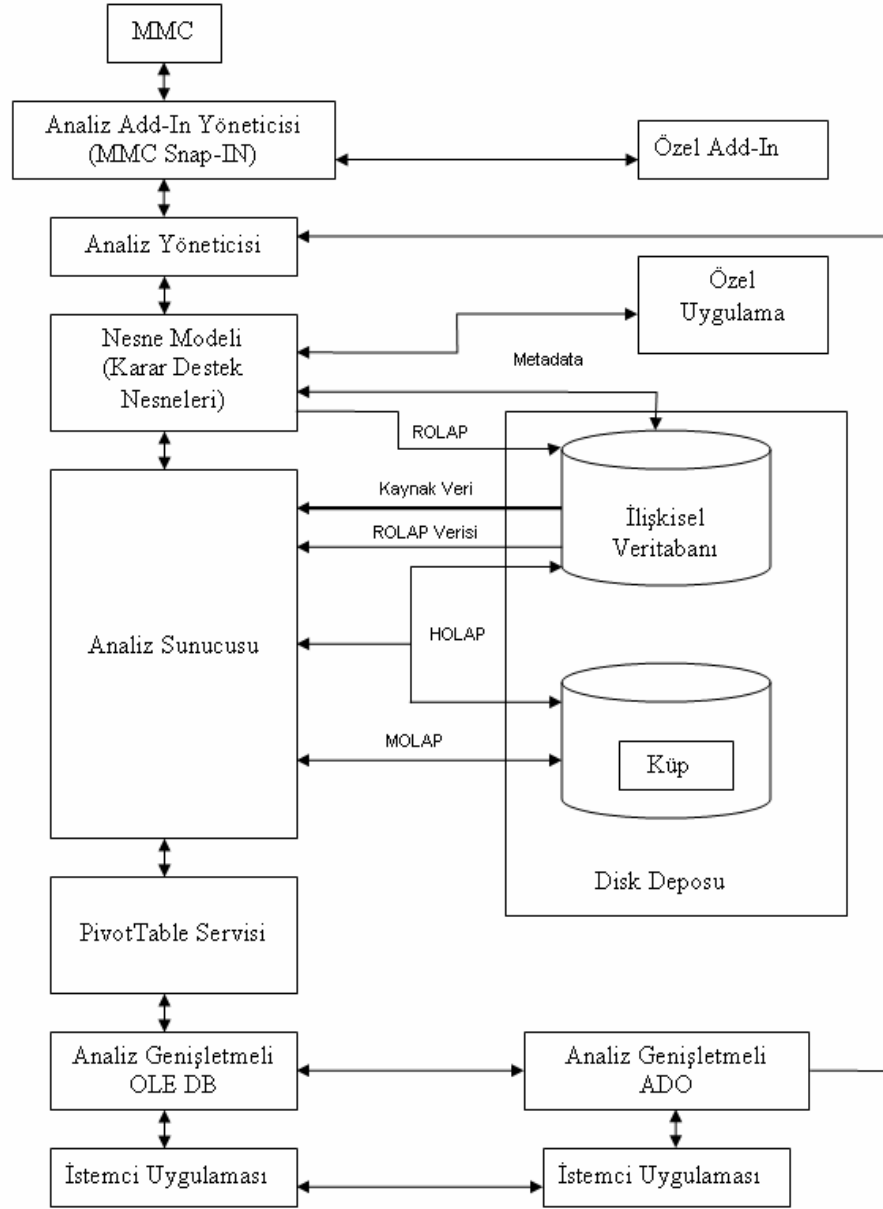
Analiz Servisi, analiz sunucusu ve PTS'ni içermektedir. Şekil 5.1'de Analiz Servisi sistem mimarisi görülmektedir. Analiz sunucusu OLAP için çok boyutlu veri küplerini oluşturmakta ve yönetmektedir. Ayrıca OLAP sağlayıcı servisleri için ADO MD ve OLE DB yoluyla bu verinin istemcilere iletilmesinde PTS için çok boyutlu veri sağlanmaktadır.

Sunucu, küp meta verisini bir havuzda depolamaktadır. Tamamlanmış küpler çeşitli veri depolama yöntemleri ile depolanabilmektedir. Çok boyutlu veritabanı dosyaları (MOLAP), ilişkisel veritabanında tablolar haline (ROLAP) ya da çok boyutlu veritabanı dosyaları ve ilişkisel tablolar halinde karma bir şekilde (HOLAP) depolanabilmesi mümkündür.

Çok boyutlu küpler için veri kaynağı OLAP veri ambarı sisteminde tipik olarak star veya snowflake modeline göre dönüştürülmüş veri barındıran ilişkisel tablolarda yerleştirilmiştir. Analiz Servisi, ODBC veya OLE DB kullanımı için bağlantıları destekleyen çok sayıda ilişkisel tabloyla birlikte çalışmaktadır. SQL Server 2000'in DTS özellikleri, Analiz Servisi'nin küpler oluşturmasında veri ambarının yönetilmesine izin vermektedir.

Sunucunun kontrolü Analysis Manager arayüzü sayesinde veya DSO nesne modeli kullanılarak gerçekleştirilen özel uygulamalar sayesinde olmaktadır. DSO sunucu tarafından küplerin yönetilmesi ve oluşturulmasını kontrol etmekte ve havuzdaki küp meta verisinin yönetilmesini sağlamaktadır. Nesne modeli, MMC konsoluna bir snap-in aracılığıyla kullanıcı arayüzü sağlayan Analysis Manager programı tarafından kullanılmaktadır. DSO nesne modeli sunucunun özel programsal kontrolünü sağlamak amacıyla Microsoft Visual Basic'te yazılan uygulamalar tarafından kullanılabilir. Aynı zamanda Analysis Manager kullanıcı arayüzü ile etkileşim içinde olan özel uygulamalar geliştirilmesine izin vermektedir.

Analysis Manager kullanıcı arayüzü, sunucu yöneticisi tarafından taranmak amacıyla çok boyutlu verinin elde edilmesinde PTS'ni kullanmaktadır. Şekil 5.1. de analiz servisi mimarisi yer almaktadır.



Şekil 5.1. Microsoft Analiz Servisi sistem mimarisi

5.1.3. Sunucu mimarisi

Analiz Servisi, OLAP küpleri ve veri madenciliği modellerinin oluşturulup yönetilmesinde sunucu yeteneklerini sağlamakta ve PTS aracılığıyla bu verilerin istemcilere iletilmesine izin vermektedir. Sunucu işlevleri şunları içermektedir:

- İlişkisel veritabanlarından küplerin oluşturulması ve işlenmesi.
- Küp verisinin çok boyutlu yapılarda, ilişkisel veritabanlarında veya her model kullanılarak karma olarak depolanması.
- Küplerden veya ilişkisel veritabanlarından veri madenciliği modellerinin oluşturulması.

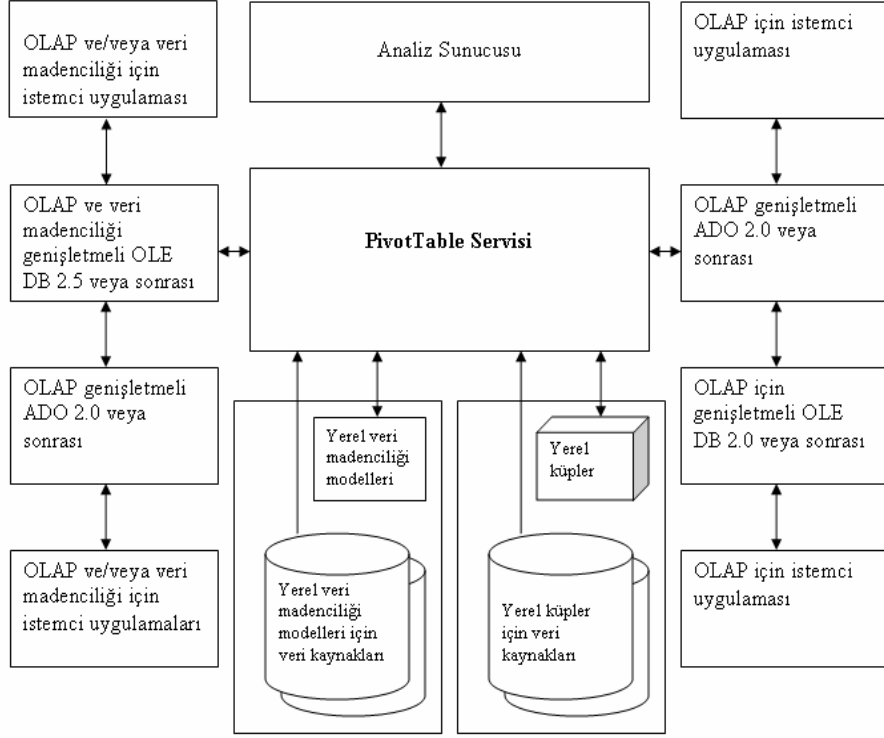
Küplerin, veri madenciliği modellerinin ve sunucu üzerindeki diğer nesnelere tanımlama bilgisi olan metadata, ilişkisel bir veritabanında metadata servisleri tarafından depolanmaktadır.

Kullanıcı arayüzü Analysis Manager tarafından sağlanmaktadır. Buna ek olarak SQL Server Enterprise Manager'ın bir parçası olan DTS, sunucu ile bir kullanıcı arayüzü sağlamaktadır. Programlama arayüzleri, Analysis Manager'da olduğu gibi sunucuyu kontrol ederek nesne modeliyle özel uygulamaların etkileşimine izin vererek sağlanmaktadır.

5.1.4. İstemci mimarisi

PTS, analiz sunucusuyla iletişim kurmakta ve istemci uygulamalarının sunucu üzerindeki OLAP ve veri madenciliği verilerine erişebilmesi amacıyla arabirimler sağlamaktadır. İstemci uygulamaları PTS'ne C++ için OLE DB arabirimleri veya ADO nesne modeli kullanarak bağlanmaktadır.

Aynı zamanda PTS, sunucu üzerindeki bir küp içerisinde veya OLE DB ilişkisel veritabanlarından veri içeren yerel küp dosyaları oluşturabilmektedir. Yerel küp dosyaları istemci bilgisayarında çok boyutlu küp dosyaları halinde depolanmış olabilir. Bu nedenle yerel küplerin sorgulanabilmesi için analiz sunucusuna bağlanmak gerekmemektedir. Ayrıca yerel küpler MOLAP depolama modeline göre depolanmışlarsa, yerel küp veri kaynaklarına bağlantı gerekmemektedir. Şekil 5.2. istemci mimarisini göstermektedir.



Şekil 5.2. İstemci Mimarisi

5.1.5. Nesne mimarisi

Analiz Servisi OLAP uygulamaları geliştirmek için çok çeşitli nesnelere sağlamaktadır. Bu nesnelere şu şekilde sıralamak mümkündür:

- **Analiz sunucusu:** Analiz sunucusu Analiz Servisi'nin sunucu bileşenini oluşturmaktadır. Analiz sunucusu çok boyutlu veri yapılarının oluşturulması, sürdürülmesi ve istemci sorguları için çok boyutlu veri sağlanması amacıyla tasarlanmıştır. Analiz sunucusu nesne hiyerarşisinde kök nesnedir. Bir kök nesne olarak nesne hiyerarşisinde ilk olarak analiz sunucusu nesnesi yaratılmaktadır. Diğer bütün Analiz Servis nesnelere sunucu bileşenin altında yer almaktadır. Analiz sunucusunu ve diğer tüm nesnelere yönetmek için kullanılan temel araç, kapsamlı bir kullanıcı

arayüzü sağlayan Analysis Manager'dır. Her sunucu analiz sunucusu havuzu olarak adlandırılan bir depolama havuzuna sahiptir. Bu havuz analiz sunucusu üzerinde tanımlanan nesnelere ait metadata'yı depolamaktadır. Her sunucu, analiz sunucusu üzerinde tanımlanmış çok boyutlu yapıları depolayan ilişkilendirilmiş bir veri klasörüne sahiptir. Bu yapılar nesnelere işlendiğinde oluşturularak dosyalar içerisinde saklanmaktadır.

- Veritabanları: Bir veritabanı küpler ve onların paylaştığı nesnelere için bir kaptır. Bu nesnelere veri kaynakları, paylaşılan boyutlar ve veritabanı rollerinden oluşmaktadır. Bu nesnelere çoklu küpler arasında paylaştırılmış ise nesnelere ve küpler aynı veritabanı içerisinde yer almalıdır. Veritabanları nesne hiyerarşisinde sunucunun hemen altında yer almaktadır.
- Veri kaynakları: Bir veri kaynağı küp gibi bir nesnenin gerek duyduğu kaynak veriye erişim için gerekli olan bilgiyi içermektedir. Veri kaynakları veri bağlantı özellikleri diyalog kutusundan oluşturulmaktadır.
- Boyut: Boyut seviyeleri olarak bilinen kategorilerin düzenlenmiş bir hiyerarşisidir. Boyutlar kullanıcının istediği bir analize temel oluşturulan benzer üyeler kümesini tanımlamaktadır. Boyutlar bir küpenin temel bileşenleridir.
- Seviye ve üyeler: Bir seviye boyut hiyerarşisinde bir elemandır. Seviyeler verinin yüksek özetlenmiş seviyesinden düşük özet seviyesine göre bir hiyerarşi içerisinde tanımlanmaktadır. Seviyeler sadece boyut içerisinde mevcuttur. Bir boyut tablosunda veya bir boyutta üye özelliklerinde sütunlara yerleştirilmişlerdir. Seviyeler boyut sihirbazı, boyut editörü veya küp editöründe bir boyut oluşturulduğunda oluşturulmaktadır.
- Ölçekler: Ölçek, küpenin olay tablosunda bir sütuna yerleştirilmiş olan bir değerler kümesidir ve genellikle nümeriktir. Son kullanıcıların küpe taradığında ilgilendikleri birincil derecede veridir.
- Hücreler: Hücre bir küpenin en küçük elemanıdır. Her küp boyutunun kesiştiği yerde bir hücre oluşmaktadır. Bir küp ölçekler, seviyeler ve boyutlar tarafından düzenlenmiş olan hücrelerden oluşmaktadır.

- Küpler: Bir küp boyut ve ölçekleri içeren çok boyutlu bir yapıdır. Ölçekler, son kullanıcının ilgilendiği nümerik değerleri sağlarken, boyutlar küpün yapısını tanımlamaktadır.
- Bölümler: Bölümler, önceden hesaplanan kümelemelerin depolanmasında ve yönetilmesinde kullanılmaktadır. Bölümler kaynak veri ve bir küpün kümeleneş verisinin çok sayıdaki sunucu makinelere dağıtılmasına izin vermektedir. Küpte yer alan her bölüm farklı bir veri kaynağına sahip olabilir. Her bölümdeki kümeleme verisi bölümlemenin tanımlandığı analiz sunucu makinesi üzerinde, başka bir analiz sunucu makinesinde veya aynı veritabanı üzerinde depolanabilmektedir.
- Kümelemeler: Kümelemeler sorgu yanıt zamanını hızlandırmak için önceden hesaplanmış yapılardır.
- Roller: Kullanıcıların istemci uygulamalarından analiz sunucusuna bağlanırken küp verisi veya veri madenciliği modellerine erişimini kontrol etmektedir.
- Komutlar: Bir istemci bir veritabanı, küp veya role eriştiğinde otomatik olarak gerçekleştirilen yönetici tanımlı yapılardır. Komutlar seçime bağlıdır. Genellikle küp genişletmelerinde yaygın olarak kullanılmaktadır. Komutlar, bir istemci uygulamasını zenginleştirmek veya küp üzerinde ilave operasyonlar gerçekleştirilmesi için son kullanıcılara izin vermektedir.
- Üye özellikleri: Boyut üyesinin bir niteliğidir. Son kullanıcılara üye hakkında ek bilgiler sağlamaktadır. Bir üye özelliğı, üye özelliğinin kullanılacağı üyeleri içeren seviyede oluşturulmaktadır.

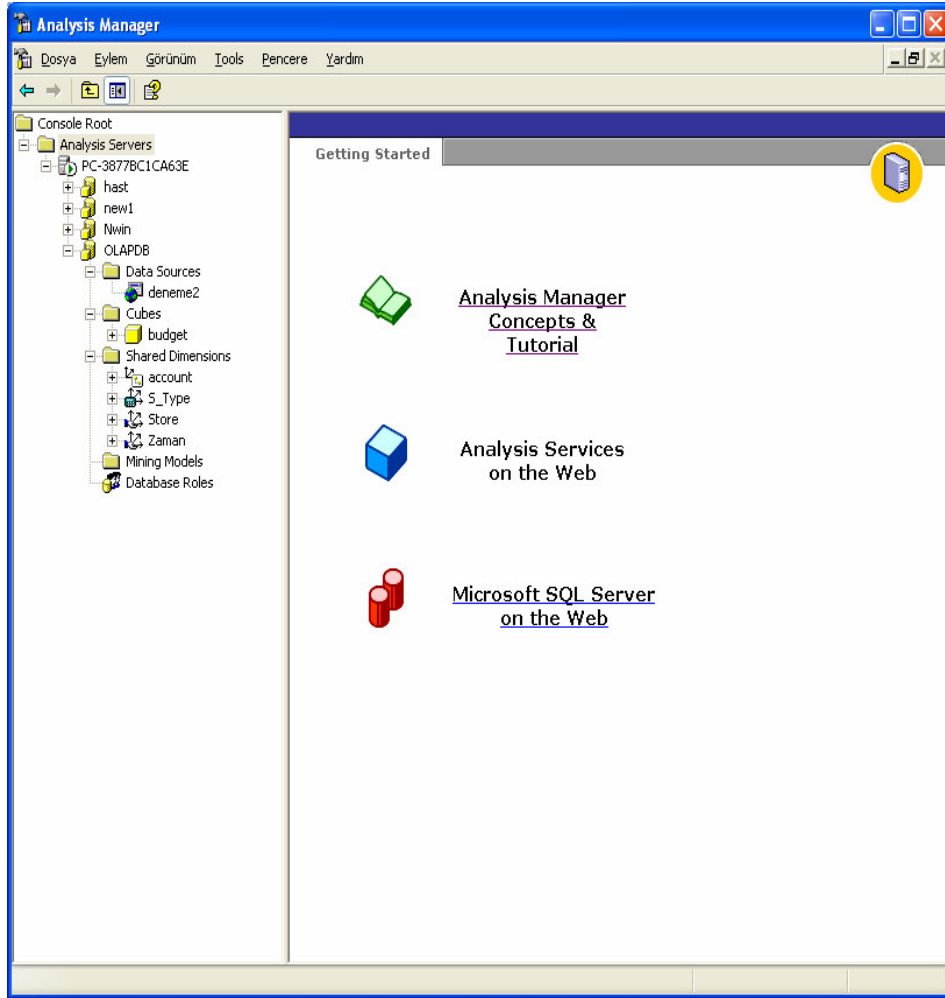
5.1.6. Analiz Servisi'nde kullanılan araçlar

Analiz Servisi'nin kullanımında yardımcı olan araçların başında Analysis Manager ve DTS gelmektedir.

5.1.6.1. Analysis Manager

Analysis Manager kullanıcıların Analiz Servisi ile etkileşim kurmasını sağlayan grafiksel araçtır. Bu araç yardımıyla yeni OLAP ve veri madenciliğı modelleri, küpler ve boyutlar oluşturulabilmektedir. Aynı zamanda farklı

kaynaklardan OLAP veritabanları ve küpler alabilmek ve onları düzenleyebilmek mümkündür. Şekil 5.3.'de Analysis Manager genel görüntüsü yer almaktadır. Analysis Manager iki temel pencereye sahiptir. Sol pencere Analiz Servisi'nde nesnelerin hiyerarşisini gösteren bir ağaç yapısı penceresidir. Sağ pencere ise seçilen nesnenin ayrıntılarını içeren bir web sayfası penceresidir.



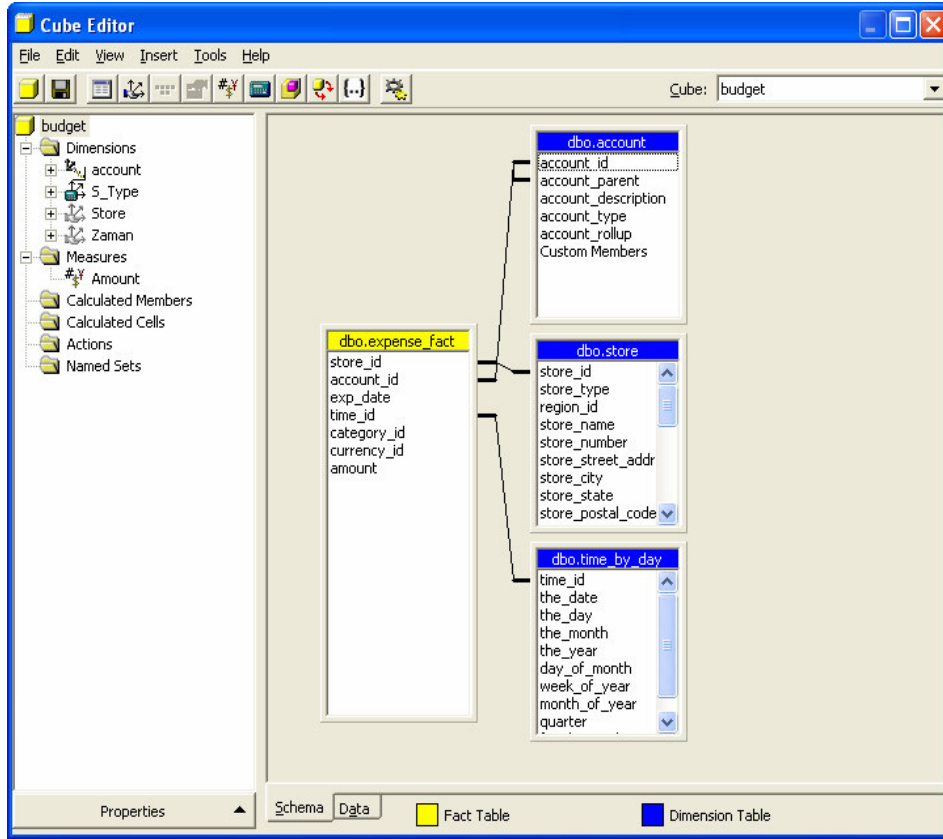
Şekil 5.3. Microsoft SQL Server 2000 Analysis Manager

Analysis Manager çeşitli görevlerin gerçekleştirilmesini daha kolay bir hale getirmek için birçok sihirbaz içermektedir. Örneğin yeni bir veritabanına bağlantı sağlamak için ya da küp ve boyutların oluşturulması için sihirbazlara sahiptir. Bu sihirbazlardan bazıları şunlardır:

- Cube wizard: Ağaç yapısı penceresinde küp nesnesine sağ tıklayarak new cube seçeneğini seçmek yoluyla bu sihirbazı çalıştırmak mümkündür. Bu sihirbaz sayesinde küp oluşturulması için gerekli olan tüm yapılar oluşturulabilmektedir.
- Dimension wizard: Boyut sihirbazı bir küp için gerekli olan paylaşılan boyutların oluşturulmasına izin vermektedir. Boyut sihirbazı küp sihirbazı içinden çalıştırılabilmekle birlikte ağaç yapısında shared dimension nesnesine sağ tıklayarak new dimension seçeneğini seçerek de çalıştırılabilmektedir.
- Storage design wizard: Küp verisinin depolama biçimini belirlemekte ve kümelemelerin tasarlanmasında yardımcı olmaktadır. Uygulama ve kullanıcıların gereklerine göre sorgulama zamanı ve depolama ihtiyaçları arasındaki optimizasyona izin vermektedir.
- Incremental update wizard: Bu sihirbaz sayesinde tüm verilerin yeniden yüklenmesine gerek kalmaksızın küp içerisine dahil edilen yeni veriler işlenmektedir.
- Partition wizard: Küp içerisindeki verinin bir bölümünü içerecek yeni bölümler yapılmasına izin vermektedir.
- Action wizard: Bu sihirbaz bir küp ya da küpün belirli bir bölümüyle ilgili bir eylem oluşturmaya yardım etmektedir.
- Calculated cells wizard: Hesaplanmış hücreler oluşturmaya yardım eder. Bir küp içerisinde alt küpler tanımlanabilmesine izin vermektedir. Hesaplanmış hücreler bir ya da daha fazla hücreyi etkileyebilmektedir.

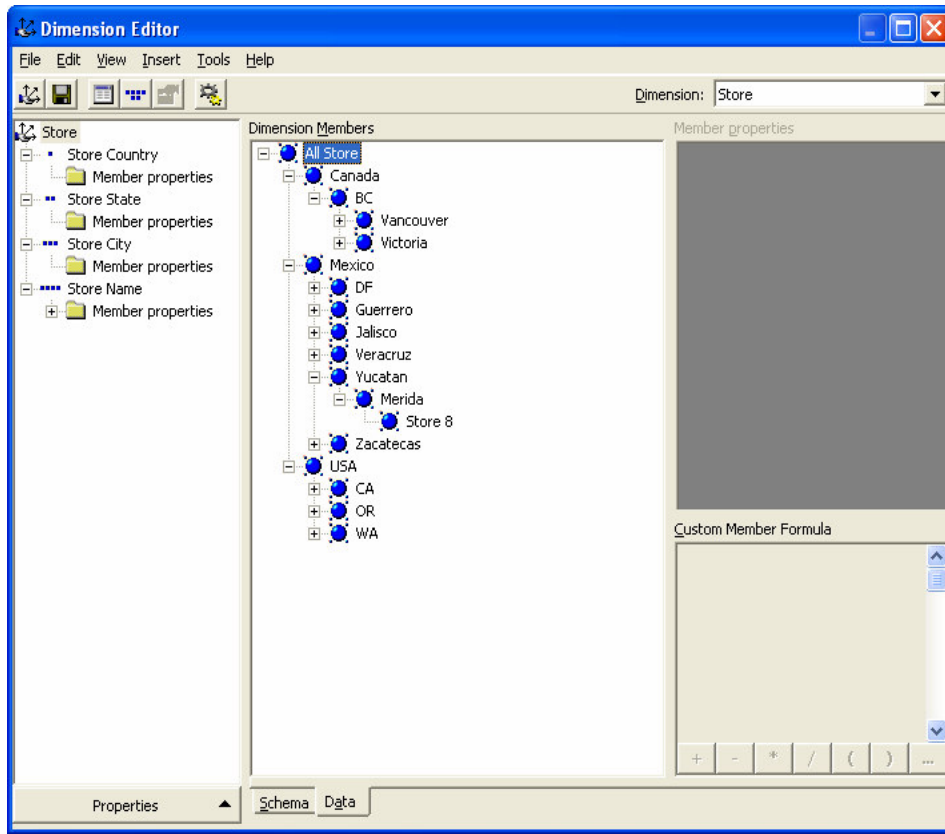
Analysis Manager'ın sahip diğer bir önemli araç küp editörüdür. Küp editörü sayesinde bir küp oluşturmak, düzenlemek ve yönetmek mümkün olmaktadır. Şekil 5.4.'te Anaysis Manager küp editörü görülmektedir. Küp editörü iki temel pencereyi içermektedir. Sol pencerede boyutlar, ölçekler, eylemler,

hesaplanmış üyeler için hiyerarşiler mevcuttur. Sağ taraf penceresi şema ve veri seçenekleri olmak üzere iki adet seçenek sunmaktadır. Şema penceresi veri ambarı veya datamart şeması bir varlık-ilişki diyagramını göstermek veya oluşturmak amacıyla kullanılmaktadır. Küp editörüne, küp üzerinde sağ tıklayarak ve edit seçeneğini seçerek ulaşmak mümkündür. Veri seçeneği seçildiğinde ise küp verisi sayfada görüntülenmektedir. Küp editörü tarafından sağlanan olanaklar arasında küp yapısının onaylanması, veri üzerinde drill down ve roll up işlemlerinin yapılması, küp depolama yapısının seçilmesi, şemanın optimizasyonu ve verinin işlenmesini saymak mümkündür. Bu fonksiyonlara küp editörünün araçlar menüsünden ulaşılabilmektedir.



Şekil 5.4. Küp editörü

Analysis Manager arayüzünde küp editörü gibi boyutların düzenlenmesi için boyut editörü mevcuttur. Yeni boyutların oluşturulmasında veya varolan boyutların düzenlenmesi ve işlenmesinde kullanılmaktadır. Paylaşılan boyutlar nesnesine sağ tıklayarak açılan pencerede new dimension ve editör seçenekleri seçilerek boyut editörüne ulaşılabilmektedir. Küp editöründe olduğu gibi şema ve veri seçenekleri mevcuttur. Veri penceresi seçildiğinde mevcut boyutta bulunan veriler görüntülenmektedir. Şekil 5.5.'te boyut editörünün veri penceresi görülmektedir.



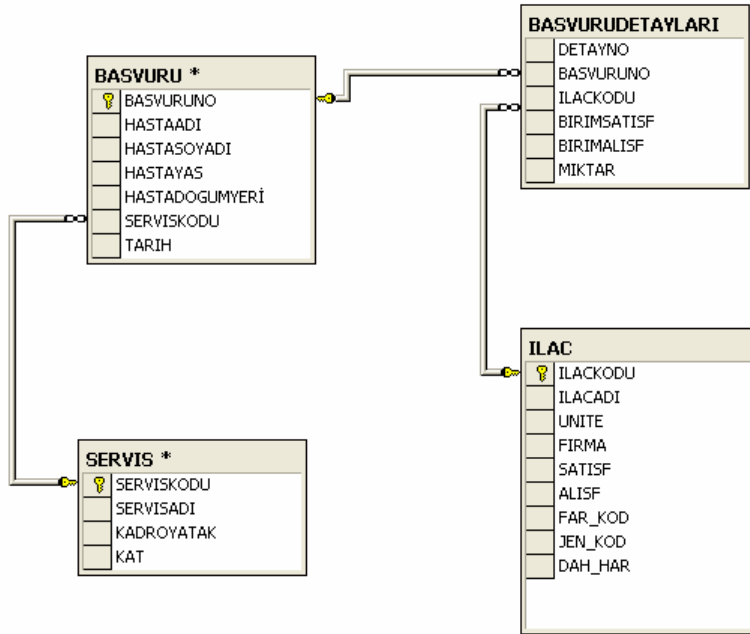
Şekil 5.5. Boyut editörü

5.1.6.2. DTS

Birçok işletmede veriler değişik biçimlerde ve farklı kaynaklarda depolanmaktadır. DTS yardımıyla farklı operasyonel sistemlerden alınan veriler seçme, dönüştürme ve birleştirme işlemleri yapılarak veri ambarı içerisine yerleştirilebilmektedir. Analiz Servisi ve veri ambarıyla doğrudan ilişkili olan DTS'e SQL Server ürünüyle birlikte gelen Enterprise Manager grafiksel aracından ulaşmak mümkündür. DTS yardımıyla işletmenin ihtiyaçlarına göre özel veri aktarım çözümleri oluşturmak mümkündür. Veri ambarları ve OLTP sistemleri arasındaki veri aktarım ve veri dönüştürme işlemleri DTS tarafından yapılmaktadır.

5.2. Örnek OLAP Uygulaması

OLAP veritabanları ve küpleri oluşturmak bir veri kaynağı gerektirmektedir. Bu uygulamada veri kaynağı olarak bir hastanenin ecza deposuyla ilgili veritabanı kullanılmaktadır. Bir küp oluşturmak için ilk adım veri kaynağının yapısını bilmektir. Oluşturulacak küpte kullanılacak veritabanı kaynağının ilgili bölümlerinin yapısı Şekil 5.6.'da gösterilmektedir.



Şekil 5.6. Eczane veri kaynağı

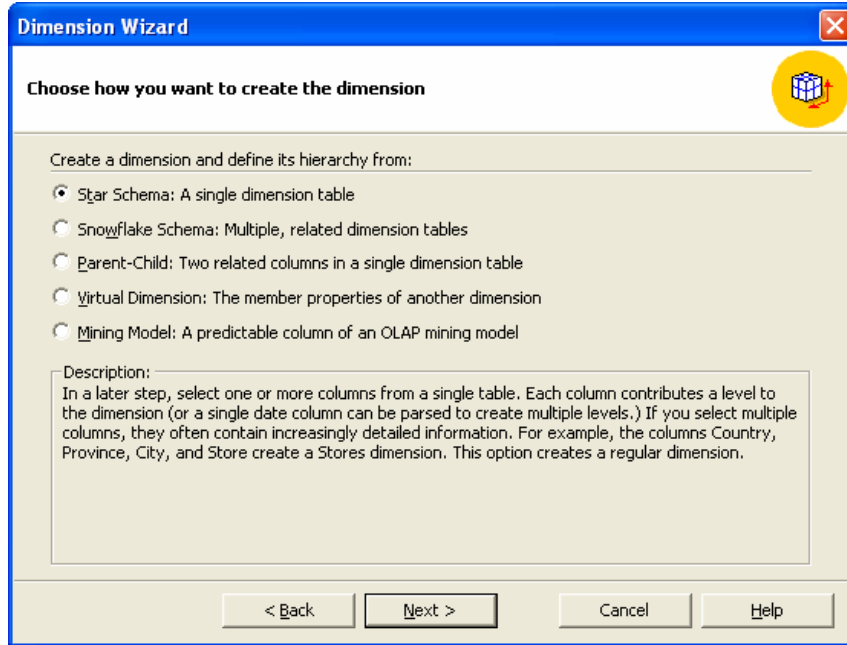
5.2.1. K p boyutlarının oluřturulması

Boyutlar SQL'deki GROUP BY ifadesine benzemektedir. OLAP k p verisi bir boyut ierisinde bilginin gruplanması veya d zenlenmesidir. K p oluřturulmadan  nce k pte yer alacak boyutlar oluřturulmalıdır. Yapılan uygulamada k p boyutları olarak zaman, servis ve ila boyutları kullanılacaktır. Servislerin zamana baėlı olarak kullandıkları ila miktarları ve gelir gider durumları analiz edilmek istenmektedir.

5.2.1.1. Zaman boyutu

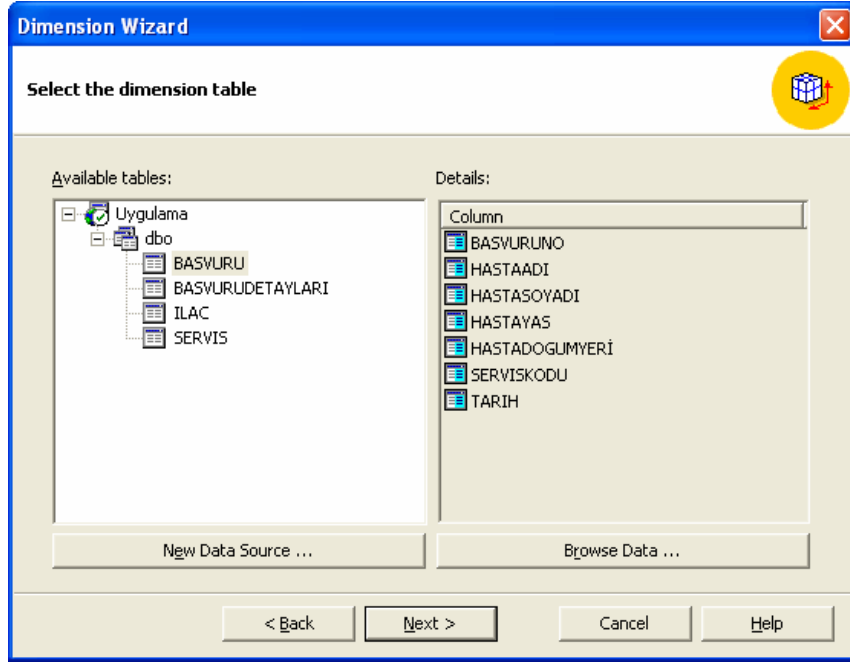
Kullanıcılar veriyi zamana baėlı olarak  zel bir hiyerarři ierisinde g rmek istemektedirler. Verinin zamana baėlı olarak analizinin gerekleřtirilmesi iin bir k pte zaman boyutunun oluřturulması gerekmektedir. Zaman boyutu boyut sihirbazı kullanılarak oluřturulmaktadır. Boyut sihirbazı aıldığında kullanıcının karřısına Őekil 5.7.'de g r len iletiřim kutusu gelmektedir.

Zaman boyutu yıldız modeline g re oluřturulacaktır. Bunun iin Őekil 5.7'de g r len iletiřim kutusunda yıldız modeli seeneėi seilmelidir. Bu seim yapıldıktan sonra Őekil 5.8.'de olduėu gibi boyut tablosu seimi iletiřim kutusundan zaman bilgisinin yer aldıėı tablo seilerek zamanla ilgili veriler alınmaktadır.

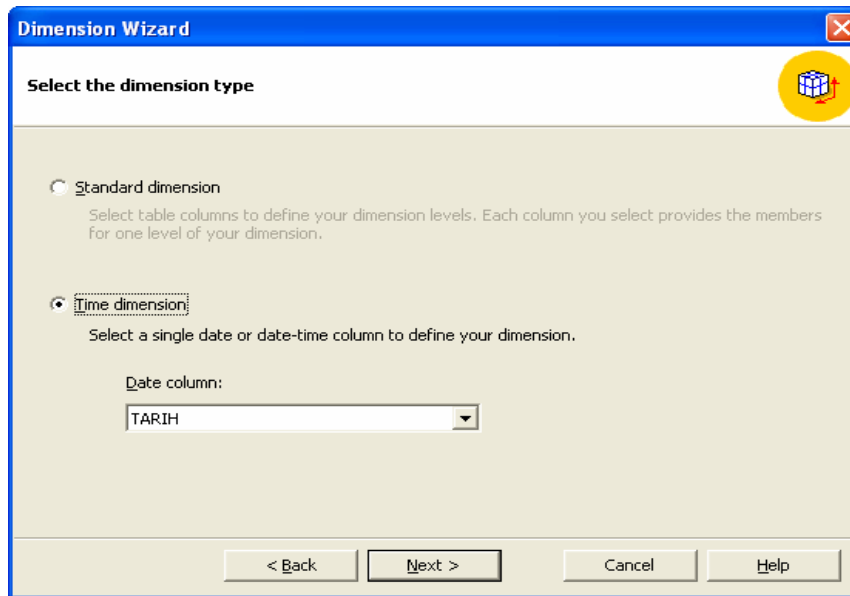


Őekil 5.7. Boyut sihirbazı iletiřim kutusu

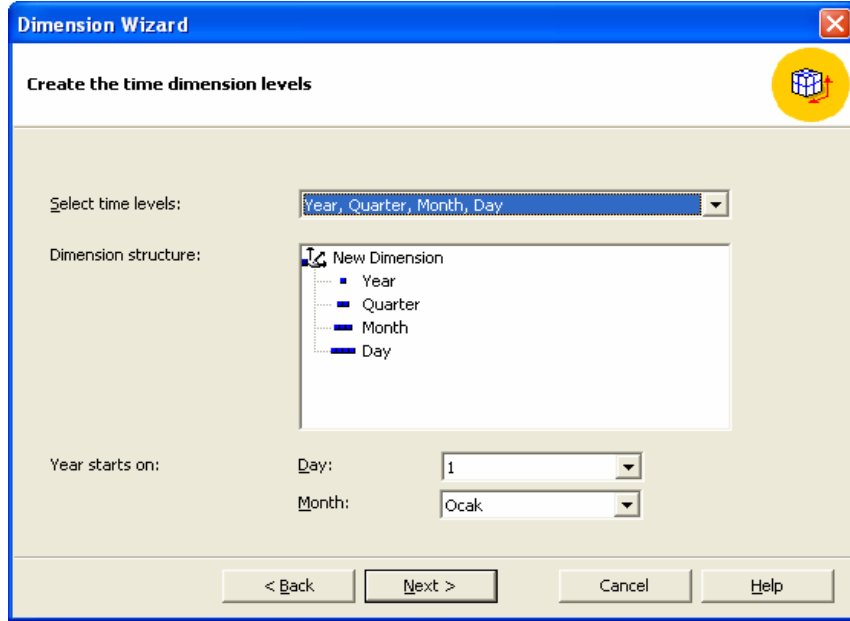
Zaman boyutu özel bir boyut olarak seçilebilmekte ve boyutla ilgili hiyerarşiler Analiz Servisi tarafından otomatik olarak oluşturulabilmektedir. Şekil 5.9.'da boyut tipi olarak zaman boyutu seçeneği seçilmelidir. Şekil 5.10.'da zaman boyutu hiyerarşi yapısı 5.11'de ise Boyut üyeleri görülmektedir.



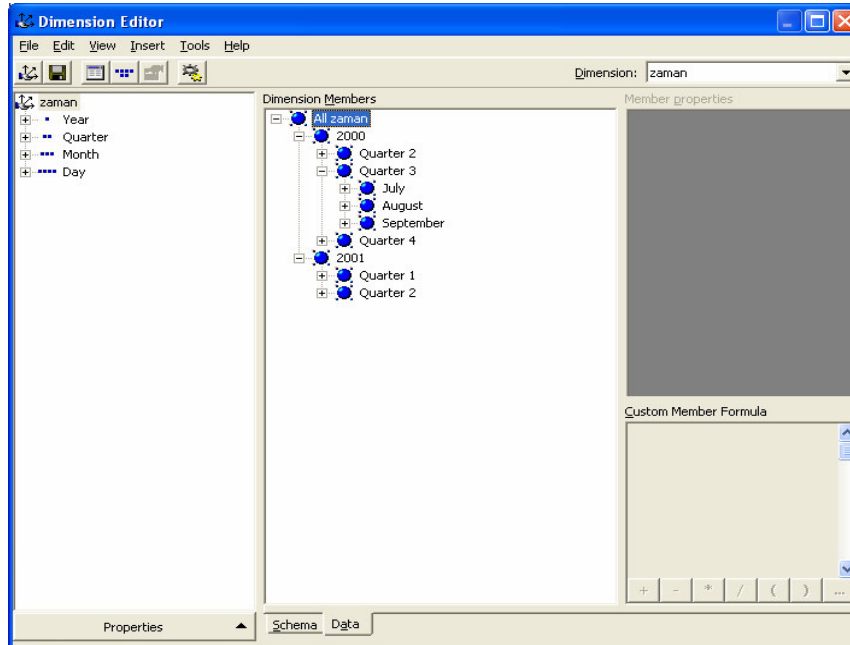
Şekil 5.8. Zaman boyutu için tablo seçimi



Şekil 5.9. Boyut tipinin belirlenmesi



Şekil 5.10. Zaman boyutunda hiyerarşi yapısı



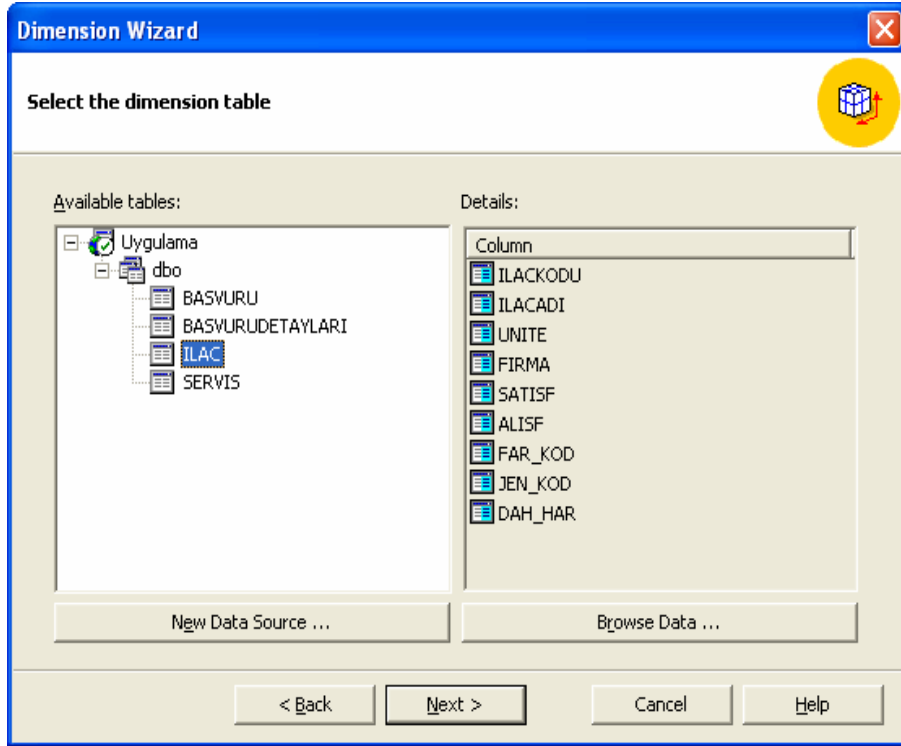
Şekil 5.11. Zaman boyutunda boyut üyeleri

Şekil 5.10'da görüldüğü gibi zaman boyutunda hiyerarşiler yıl, çeyrek, ay ve gün olarak belirlenmiştir. Oluşturulan zaman boyutunda boyut üyeleri ise Şekil 5.11'de görülmektedir.

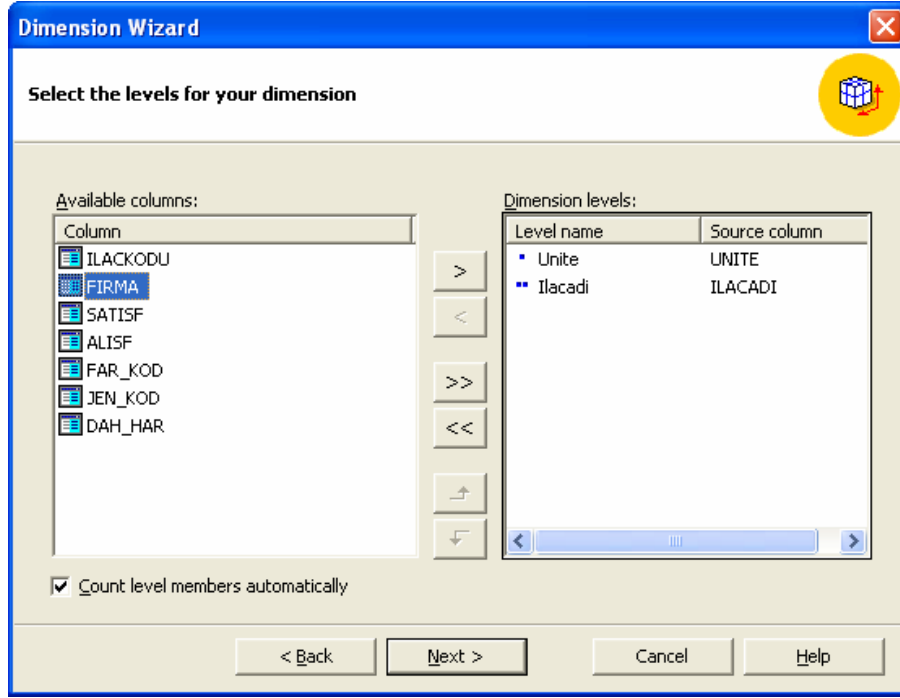
5.2.1.2. İlaç boyutu

Şekil 5.12.'de ilaç boyutunun oluşturulması için gerekli tablonun seçimi ve tablo yapısı görülmektedir. İlaç boyutu zaman boyutunda olduğu gibi yıldız modeli seçilerek oluşturulmaktadır.

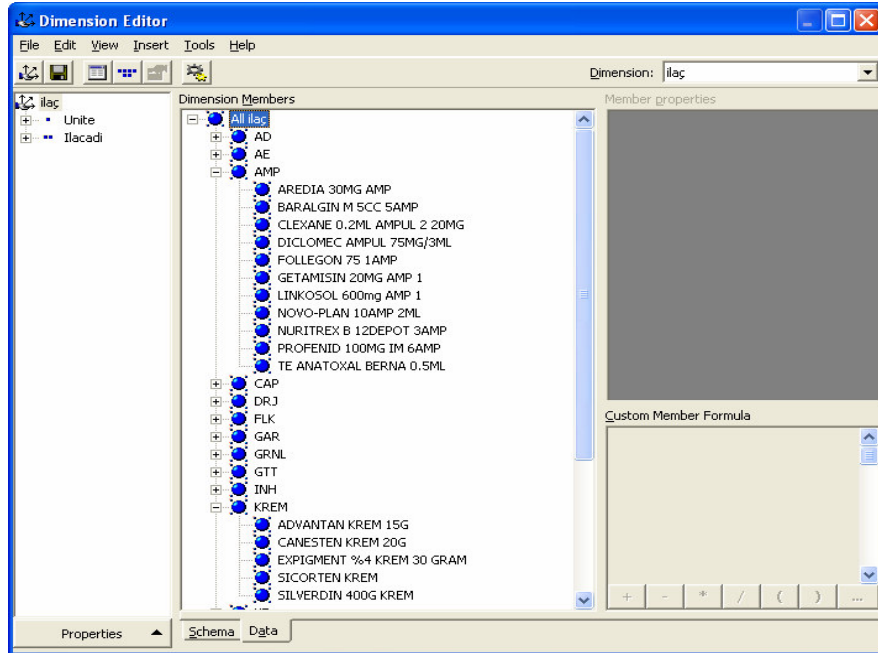
Zaman boyutunda boyut seviyeleri Analiz Servisi tarafından otomatik olarak oluşturulmaktadır. Standart boyut oluşturulmasında ise boyut seviyeleri kullanıcı tarafından belirlenmektedir. Şekil 5.13.'de boyut seviyeleri olarak ünite ve ilacadi belirlenmiştir. Şekil 5.14'de ise ilaç boyutunda boyut üyeleri yer almaktadır.



Şekil 5.12. İlaç boyutunda tablo seçimi



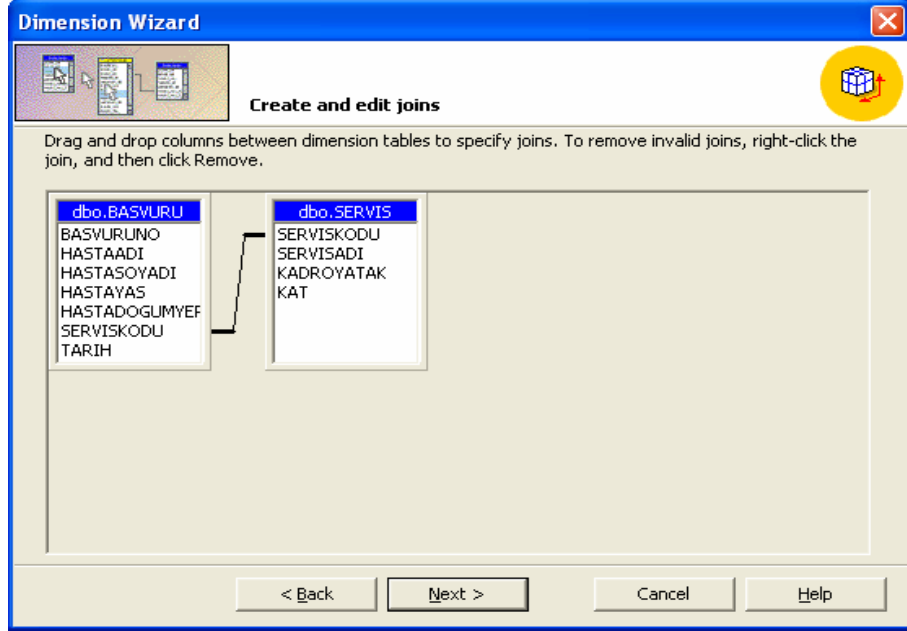
Şekil 5.13. İlaç boyutunda seviyelerin oluşturulması



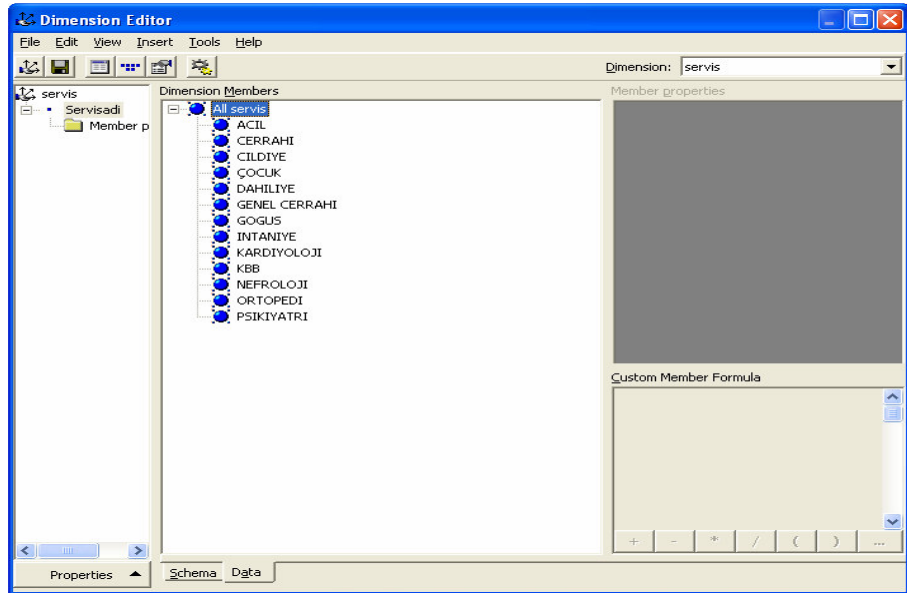
Şekil 5.14. İlaç boyutunda boyut üyeleri

5.2.1.3 Servis boyutu

Servis boyutunun oluşturulmasında Şekil 5.15.'de ki gibi ilişkili iki tablodan yararlanılmaktadır. Bu yüzden snowflake şema modeli seçilmesi gerekmektedir. Servis boyutuna ait boyut üyeleri Şekil 5.16.'da görülmektedir.



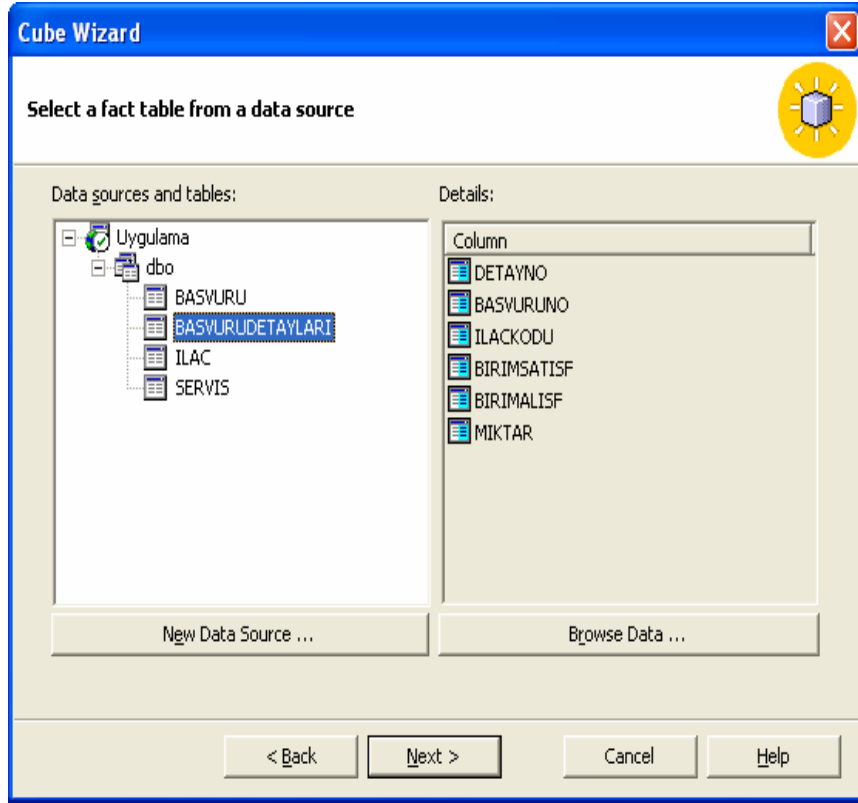
Şekil 5.15. Servis boyutunda kullanılan tablolar



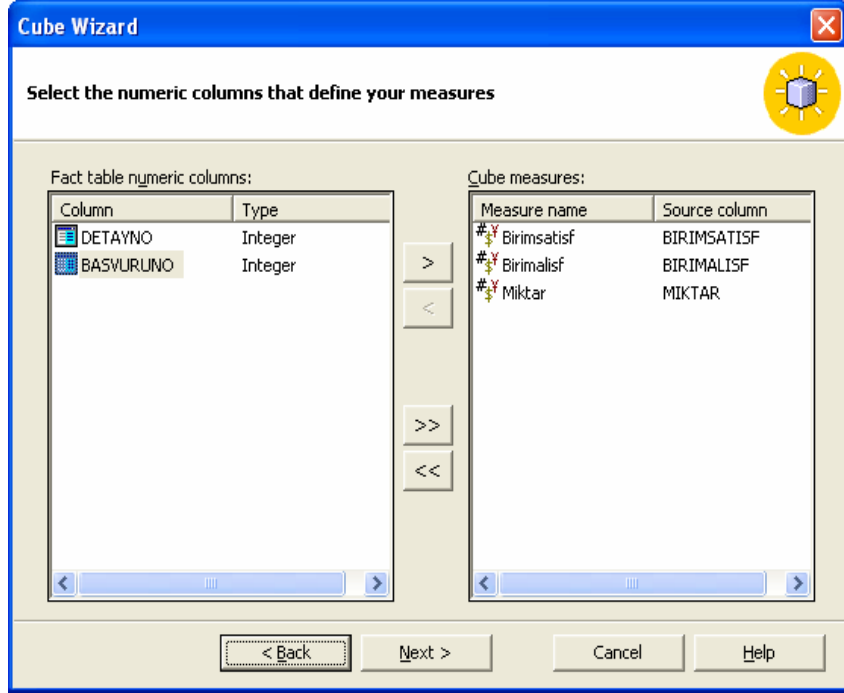
Şekil 5.16. Servis boyutunda boyut üyeleri

5.2.2. Küp oluşturulması

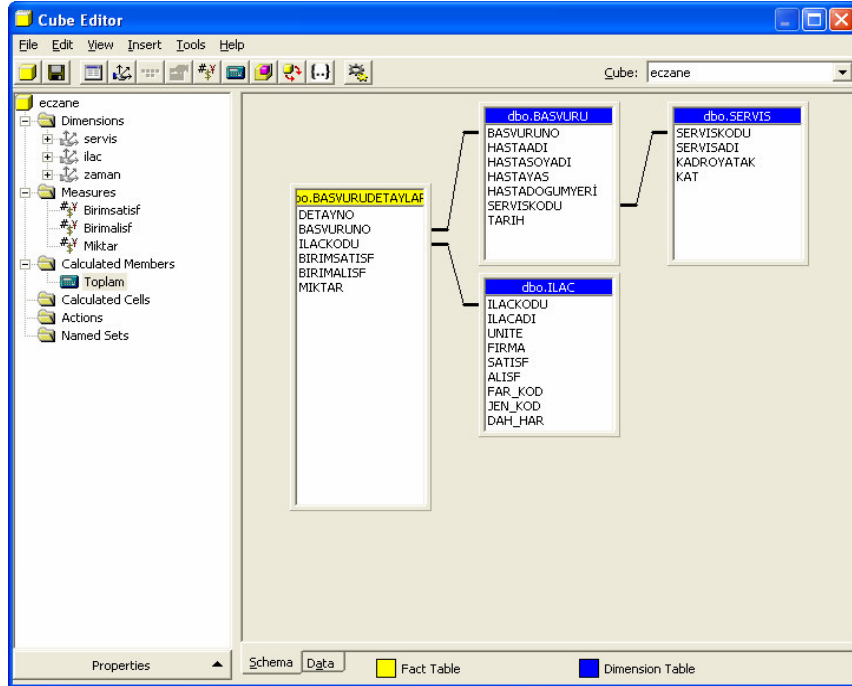
Küp için gerekli boyutlar oluşturulduktan sonra küp sihirbazı yardımıyla olay tablosu ve olay tablosu ile ilişkili boyutlar seçilmektedir. Şekil 5.17’de olay tablosu olarak başvuru detayları isimli tablo veri kaynağından seçilmektedir. Analiz konusunun niceliğini belirleyen ölçekler ise Şekil 5.18.’de birim satış fiyatı, birim alış fiyatı ve kullanılan ilaç miktarı olarak belirlenmiştir. Ayrıca net geliri belirleyebilmek amacıyla toplam isimli bir hesaplanmış üye yaratılmıştır. Oluşturulan küpün yapısı küp editöründe Şekil 5.19.’da olduğu gibi görüntülenmektedir.



Şekil 5.17. Olay tablosunun seçilmesi



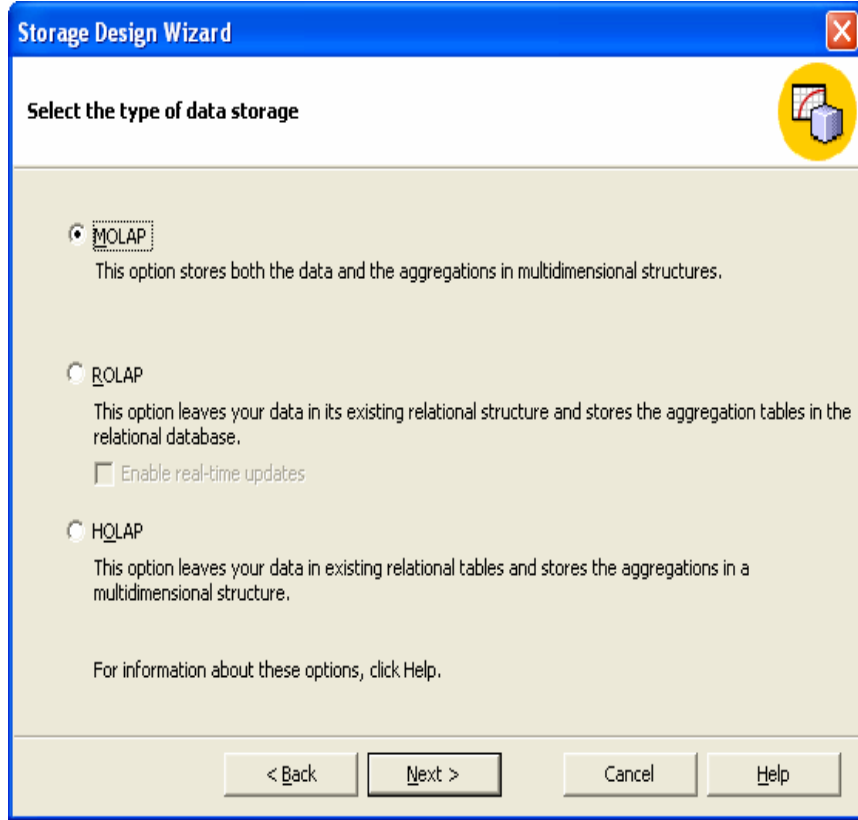
Şekil 5.18. Ölçeklerin belirlenmesi



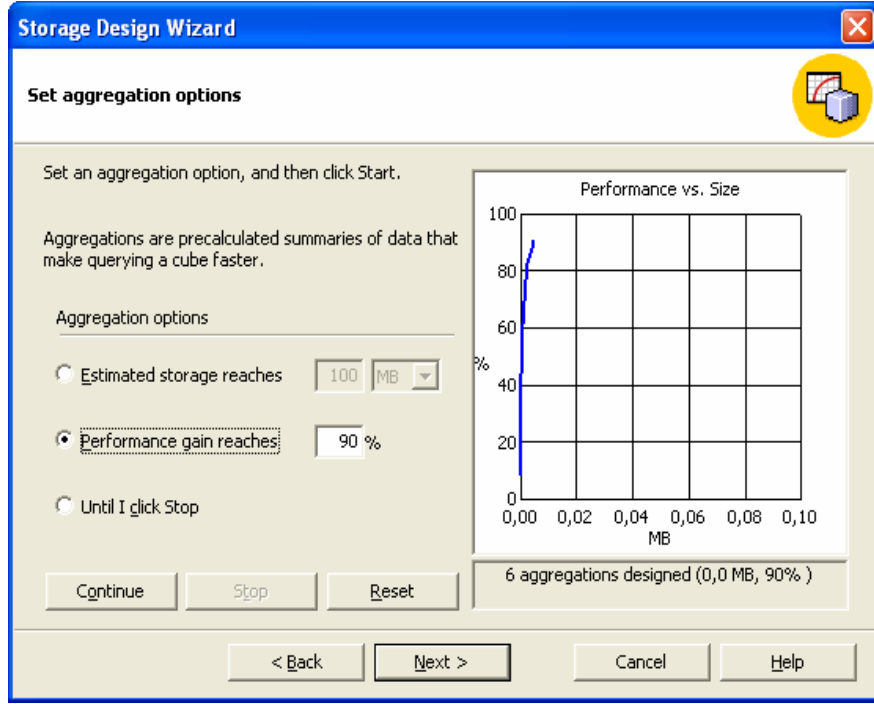
Şekil 5.19. Eczane küpünün yapısı

5.2.2.1. Depolama türünün belirlenmesi

Oluşturulan küpler depolama sihirbazı yardımıyla MOLAP, ROLAP veya HOLAP modeli kullanılarak depolanabilmektedir. Uygulamada eczane küpü için MOLAP modeli kullanılmaktadır. Bunun için Şekil 5.20.'de MOLAP seçeneği seçilerek Şekil 5.21.'de ki iletişim kutusundan kümeleme seçenekleri ayarlanmaktadır. Sorgulama performansı ve veritabanı büyüklüğü arasındaki optimizasyon bu iletişim kutusundan yapılmaktadır.



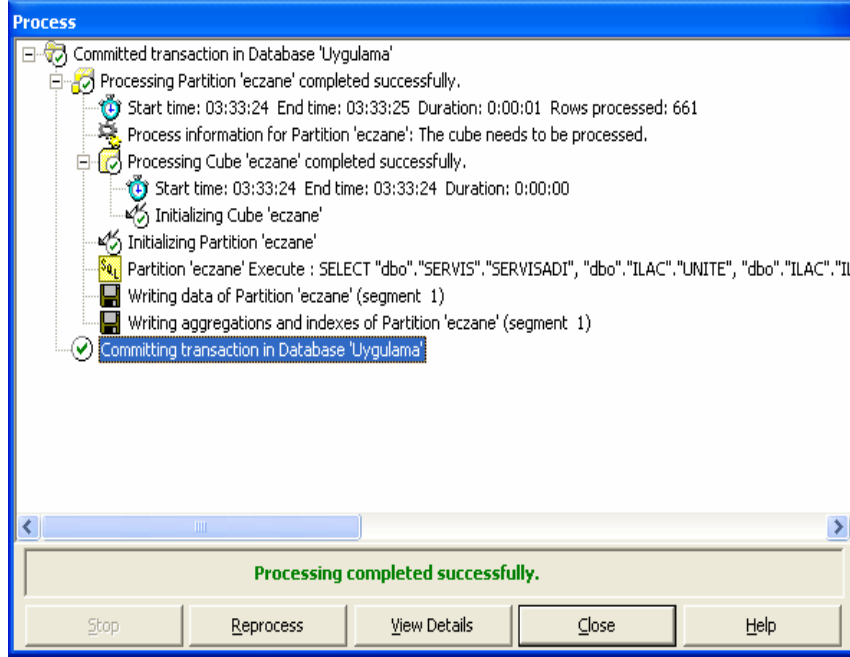
Şekil 5.20. Depolama türünün belirlenmesi



Şekil 5.21. Kümeleme seçeneklerinin ayarlanması

5.2.2.2. Oluşturulan küpün işlenmesi

Oluşturulan küpün görüntülenebilmesi ve analizler için kullanılabilmesi için işlenmesi gerekmektedir. Bu işlemi küp editöründe araçlar menüsünden küp işleme seçeneği seçilerek yapmak mümkündür. Şekil 5.22.'de OLAP eczane küpünün işlenmesi gösterilmektedir. İşlenen veri Şekil 5.23.'de olduğu gibi küp tarayıcısı kullanılarak görüntülenmekte ve hastane servislerine hangi tarihte hangi ilaçtan ne kadar gönderildiği veri üzerinde drill down ve roll up işlemleri yapılarak görülebilmektedir.



Şekil 5.22. OLAP küpünün işlenmesi

Measures Miktar

		Servisadi	+ Year		CERRAHI		
		ACIL	All zaman		+ 2000	+ 2001	All
- Unite	Ilacadi	+ 2001					
All ilac	All ilac Total	281,00	181,00	107,00	74,00		
+ AD	AD Total	4,00					
+ AE	AE Total	17,00	12,00	12,00			
	AMP Total	18,00	28,00	6,00	22,00		
	AREDIA 30MG AMP	2,00					
	BARALGIN M 5CC 5AMP	4,00					
	CLEXANE 0.2ML AMPUL 2 Z	2,00					
	DICLOMEC AMPUL 75MG/3	2,00	21,00			21,00	
	FOLLEGON 75 1AMP	8,00					
- AMP	GETAMISIN 20MG AMP 1						
	LINKOSOL 600mg AMP 1						
	NOVO-PLAN 10AMP 2ML						
	NURITREX B 12DEPOT 3AN						
	PROFENID 100MG IM 6AMI						
	TE ANATOXAL BERNA 0.5N		7,00	6,00	1,00		
+ CAP	CAP Total	15,00	8,00		8,00		
	DRJ Total	1,00					
- DRJ	EPIDOSIN COMP 20 DRJ	1,00					
	UTERJIN DRAJE 20						
+ FLK	FLK Total	38,00	25,00	21,00	4,00		
+ GAR	GAR Total						
+ GRNL	GRNL Total	12,00	1,00		1,00		

Double-click a member to drill up or down.

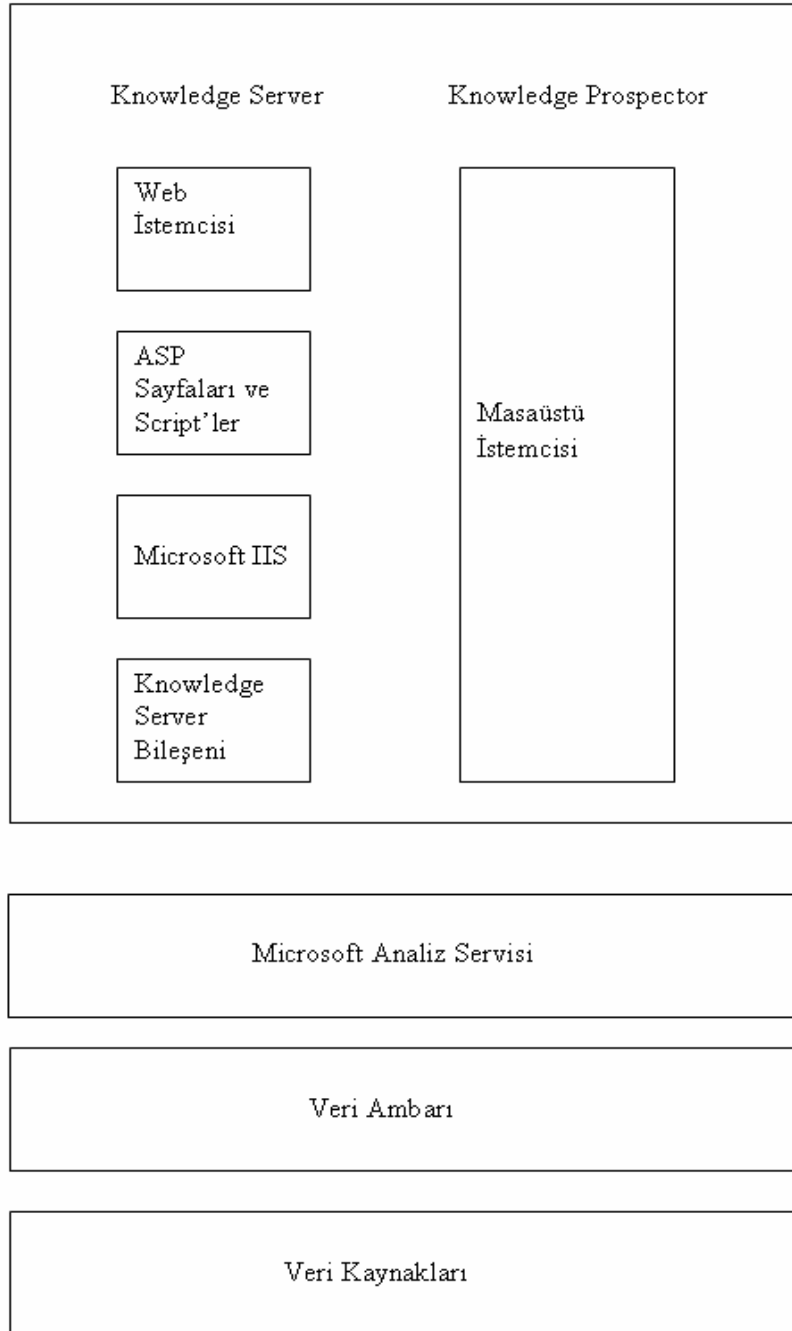
Close Help

Şekil 5.23. Eczane küp verisi

5.3. İstemci Uygulaması Olarak Cubularity Knowledge Platform Yazılımı

Knowledge Platform var olan operasyonel sistemler, veri ambarları ve Microsoft OLAP yeteneklerini temel almaktadır. Organizasyon üyelerinin ihtiyaçlarına uygun analiz, rapor, karar destek veya iş zekası çözümleri sağlamaktadır. Knowledge Platform, güçlü ve esnek OLAP çözümleri üretilmesi konusunda Microsoft SQL Server 2000 Analiz Servisleri'nin yeteneklerini artırmaktadır.

Knowledge Platform'un en önemli özelliği bir analiz raporunu oluşturma, kaydetme ve yükleme özelliğidir. Rapor bir OLAP küpü ve onun boyut ve hiyerarşilerinden türetilmektedir. Özel bir analiz boyutlar, her boyut için belirtilen hiyerarşiler, her hiyerarşiden seçilen üyeler ve çok sayıda görünümünü belirtmektedir. Bir görünüm boyutların seçilip bir eksen üzerinde satır ve sütunlar şeklinde yerleştirilmesi sonucu oluşturulmaktadır. Analizlerin bu şekilde oluşturulması, kaydedilmesi ve paylaşılması organizasyon üyelerinin yapmış oldukları çalışmaları aralarında paylaşmalarına ve birlikte çalışmalarına olanak sağlamaktadır. Şekil 5.24. Knowledge Platform Mimarisini göstermektedir.



Şekil 5.24. Knowledge Platform Mimarisi

5.3.1. Knowledge Platform bileşenleri

Knowledge Platform, analizlerin yapılması ve bu analizlerin paylaşılması için Knowledge Prospector ve Knowledge Server olmak üzere iki bileşenden oluşmaktadır.

5.3.1.1. Knowledge Prospector

Knowledge Prospector veriye hızlı bir şekilde erişilmesi ve analiz edilmesinin yanı sıra raporların isteğe göre uyarlanması ve dağıtılmasını da sağlamaktadır. Veri, hesaplanmış veri tablolarında sunulmakta ve ileri analizler için iki ve üç boyutlu çizim ve grafikler şeklinde gösterilebilmektedir. Knowledge Prospector, Microsoft Analiz Servisi veri küplerinden bilginin elde edilmesi, hızlı ve kolay bir şekilde veriye erişilmesi, analiz edilmesi, sunulması ve paylaşılması için birçok yararlar ve işlevsellikler sağlamaktadır.

Yararları:

- Kolay sorgulama ve görüntüleme:
Knowledge Prospector, gerçek zamanlı analizler gerçekleştirmek ve zamana yönelik karar vermeyi sağlamak amacıyla çok sayıda olanak sunmaktadır. Analizler hızlı bir şekilde gerçekleştirilebilmekte ve kaydedilebilmektedir. Kural temelli seçme, seçme kriterinin sonraki analizlerde de kullanılmasını sağlamaktadır. Çok boyutlu analizler kolay bir şekilde gerçekleştirilmektedir. Boyutlar verinin farklı bakış açılarından incelenmesini sağlayacak şekilde yerleştirilebilmektedir. Boyut üyeleri seçilerek basit sürükle-bırak işlemiyle analize dahil edilebilmektedir.
- Artan verimlilik: Knowledge Prospector, sürükle-bırak, araç çubuğu düğmeleri ve sağ tıklama menüleri gibi alışılmış ve öğrenilmesi kolay araçlarla kullanıcıların en iyi şekilde çalışmalarını sağlamaktadır. Görüntü üzerinde verinin hızlı bir şekilde derinlemesine analizi, daha yüksek hassasiyet ve plan üzerinde veriye inme seçenekleri analizleri kolaylaştırmaktadır. Knowledge Prospector kullanıcılara aşağıdaki olanakları sağlamaktadır:
 - Drill Down/Up: Veri hiyerarşisinde bir sonraki üye seviyelerini görüntüleyebilme.

- Non-Destructive Drilling: Üye ve ebeveyn kümeleme verisinin her ikisinin de hızlı bir şekilde görüntülenebilmesi.
- Drill Through: Boyutun temel ilişkisel verisiyle ilişkilendirilmesi.
- Actions: Bir elektronik tablo, belge ya da web sitesi gibi küp, boyut, seviye veya hücreyle ilişki ek bilgilere erişim.
- Kişisel Hiyerarşiler: Kullanıcının özel ihtiyaçlarına göre yeni hiyerarşiler oluşturabilme veya kullanıcının ayrı bir klasöre özel olarak üye grupları oluşturulabilmesi. Grup ve hiyerarşiler otomatik olarak kümelenebilir.
- Güçlü analiz: Güçlü ve esnek analiz olanaklarının sağlanmasına, problemlerin, eğilimlerin ve fırsatların en etkin şekilde belirlenmesine izin vermektedir.
 - Çizelgelenmiş analiz: Tablolar bir satır veya sütun olarak herhangi bir boyuttan oluşturulmuş olabilir. Satır ve sütunlar bir boyut hiyerarşisindeki her seviyede üyeler içerebilmektedir.
 - Esnek veri görüntüleme: Çok sayıda grafik ve çizelge tipleri analizin anlaşılmasını sağlamaktadır.
 - Güçlü ve sağlam raporlama: Sonuçlar özel olarak belirtilmiş bir kritere göre filtrelenebilir renklendirilerek analizi kolaylaştırmaktadır.
- Esnek işbirliği ve paylaşım: Bugünün şartlarında sorunların kolay ve hızlı bir şekilde belirlenmesi gerekmekte ve bulunan sonuçlara çözüm getirecek ve geliştirecek diğer çalışanlarla paylaşılması gerekmektedir.
 - Knowledge Prospector analizlerinin paylaşımı: analizler kaydedilebilmekte ve ağ, e-mail veya web sayfaları olarak yayınlanarak dinamik ve statik veri olarak görüntülenerek diğer çalışanlarla paylaşılabilir.
 - Diğer uygulamalar ve web e aktarım: Tablo ve çizelgeler kolay bir şekilde yaygın olarak kullanılan Excel, Word gibi uygulamalara taşınabilmektedir.
 - Ek gösterim seçenekleri: Tek bir analiz çok sayıda görünüm içerebilmektedir. Aynı analiz, çubuk çizelgesi, veri tablosu şeklinde kaydedilebilmektedir.

İşlevsellikleri:

- Seçim: Daha çok hassasiyet ve kolaylık için Knowledge Prospector analiz için parça veya üyelerin seçiminde çok çeşitli yöntemler sunmaktadır:
 - Drag and Drop (sürükle-bırak): bir hiyerarşide herhangi bir seviyenin veya seviyedeki üyelerin seçilerek seçim alanına sürüklenip bırakılması ile analize dahil edilmesi mümkündür
 - Kural temelli seçim: Kural temelli seçim kullanıcının kendi özel kriterini belirlemesine izin vermektedir.
 - Özel klasörler ve hiyerarşiler oluşturulması: Kullanıcı var olan küp yapısını değiştirmeden kendi boyut hiyerarşilerini oluşturabilmektedir. Oluşturulan hiyerarşi klasörler, diğer hiyerarşiler, altseviyeler, özel zaman periyotları ve kural temelli seçimleri içerebilmektedir.
 - Sunucu tabanlı veya yerel küplere erişim: Sunucu tabanlı küplerden olduğu gibi yerel küplerden veriye erişime de izin vermektedir. Bu şekilde herhangi bir yerden analizleri gerçekleştirmek mümkün olmaktadır.
- Navigasyon:
 - Genel navigasyon: Veri gösteriminin her tipi için tablolar, grafikler, çizelgeler gibi çok çeşitli navigasyon araçları sağlanmaktadır.
 - Genel kontroller: Dilimleyici boyut üyesi kontrollerinin konumu veri gösterim veya görüntüleri arasında sabittir fakat her görüntü boyutların farklı kombinasyonunu içerebilmektedir.
 - Hiyerarşik navigasyon: Standart görüntü tiplerinin her birinde analiz servisindeki ilişkisel veri kaynağına drill through yapılabildiği gibi boyutun hiyerarşik yapısında drill up, drill down veya drill across operasyonları yapılabilmektedir.
 - Gizle\göster: Herhangi bir üye, maddelerin daha iyi canlandırılabilmesi için gizlenebilmekte veya gösterilebilmektedir.
 - Eylemler: Bir analiz servisi özelliğindeki eylemler desteklenmektedir.
 - Sıralama: Tek bir üye satır veya sütununda sağ tıklama komutu kullanılarak sıralama yapılabilmektedir. Karmaşık çok seviyeli sıralamalar yapabilmek için kural oluşturucusu kullanılabilir.

- Analiz ve canlandırma: Knowledge Prospector çok çeşitli veri gösterim ve canlandırma biçimlerini sağlamaktadır:
 - Veri tabloları: Verinin çizelgeli gösterimleri hücrelerdeki ölçek değerlerini göstermektedir. Boyut üyeleri satır ve sütunlarda tutulmaktadır.
 - Grafik ve çizimler: Çizgi, çubuk, alan ve üç boyutlu çubuk çizimleri şeklinde gösterilebilmektedir.
 - Çeyrek daire ve dağıtık çizimler: Çok ölçekli grafikler 2 boyutlu olarak dağıtık taslak şeklinde gösterilebilmektedir.
 - Biçimlendirme ve görüntü üzerinde navigasyon: Taslakların ve çizelgelerin çoğu boyut ve boyut üyelerinin yeniden sıralanmasına izin vermektedir. Puntolar, renkler ve ölçeklemeler çoğu görüntü içinde ayarlanabilmektedir.
 - Renklendirme: Çeşitli çizelge ve grafik biçimlerine ek olarak raporların renklendirilmesine de izin verilmektedir. Koşullara bağlı renklendirme üye ve verilerin değerlere göre renklendirilerek daha iyi anlaşılmasını sağlamaktadır.
- Sunum, kayıt ve paylaşım:
 - Ad-hoc ve standart raporlama: Knowledge Prospector, paylaşımlı ve etkileşimli raporlamayı sağlamak amacıyla olduğu kadar ad hoc analizleri kolaylaştırmak amacıyla da tasarlanmıştır. Bu tasarım ortak bir veri kaynağından analizlerin paylaşılmasına izin vermektedir. Ek olarak standart raporlamaya temel oluşturan veriyi inceleme yeteneği vardır.
 - Kaydetme ve yeniden kullanım: Tüm analiz görüntüleri ve veri seçimi belirtilen tek bir dosyaya kaydedilebilmekte ve paylaşılabilir. Knowledge Prospector bu dosyaları .c3 uzantılı dosyalar olarak oluşturmaktadır. .C3 dosyalarının paylaşımı ortak veri kaynağından elde edilmiş bilginin genel görüntüsünü sağlamaktadır.
 - Export: Knowledge Prospector, Excel, Word, HTML gibi uygulama ve biçimlerden verinin seçilerek alınması ve analiz edilmesine izin vermektedir.

5.3.1.2. Knowledge Server

Knowledge Server, sunucu tabanlı bir uygulamadır ve web temelli OLAP çözümleri oluşturmak için geliştirme platformudur. Web Renderer, örnek script'ler ve ASP sayfaları, nesne modeli içermektedir. Knowledge Server, özel çözümler oluşturmak ve internet veya intranet üzerinden veri erişimi için gerekli tüm özellikleri sağlamaktadır.

Web Renderer: Web Renderer, Knowledge Prospector ile oluşturulmuş .c3 dosyalarının intranet veya internet üzerinden paylaşılmasına izin vermektedir. Kullanıcılar görmek istedikleri analizleri web tarayıcı yardımıyla görüntüleyebilmektedir.

Web Renderer iki şekilde kullanılabilir:

- Dinamik: İstenildiği anda OLAP küpü üzerinde gerçek zamanlı sorgular yapmak mümkündür. Kullanıcı varolan veriyle etkileşim içindedir.
- Statik: Analiz dosyasının okunması için script'ler kullanılmaktadır. Sonuçlar OLAP küpünden bir kere alınmaktadır. Bu şekilde performansın daha hızlı olması sağlanmaktadır. Eğer OLAP küpüne çok sık bağlanmak gerekmiyorsa bu yöntem kullanılmalıdır.

Örnek Script ve ASP sayfaları: Knowledge Server ile kaydedilmiş analiz dosyaları üzerinde çeşitli görevleri gerçekleştirmek için Windows Script'leri kullanılarak özel script oluşturmak mümkündür. Knowledge Server kullanıcının kendi script'lerini oluşturması için örnek script'ler içermektedir.

Knowledge Server nesne modeli: Knowledge Server nesne modeli belgeleri kullanılarak geliştiricilerin Knowledge Server ile birlikte çalıştırılabilen özel script, web sayfaları ve uygulamalar oluşturması mümkündür [58].

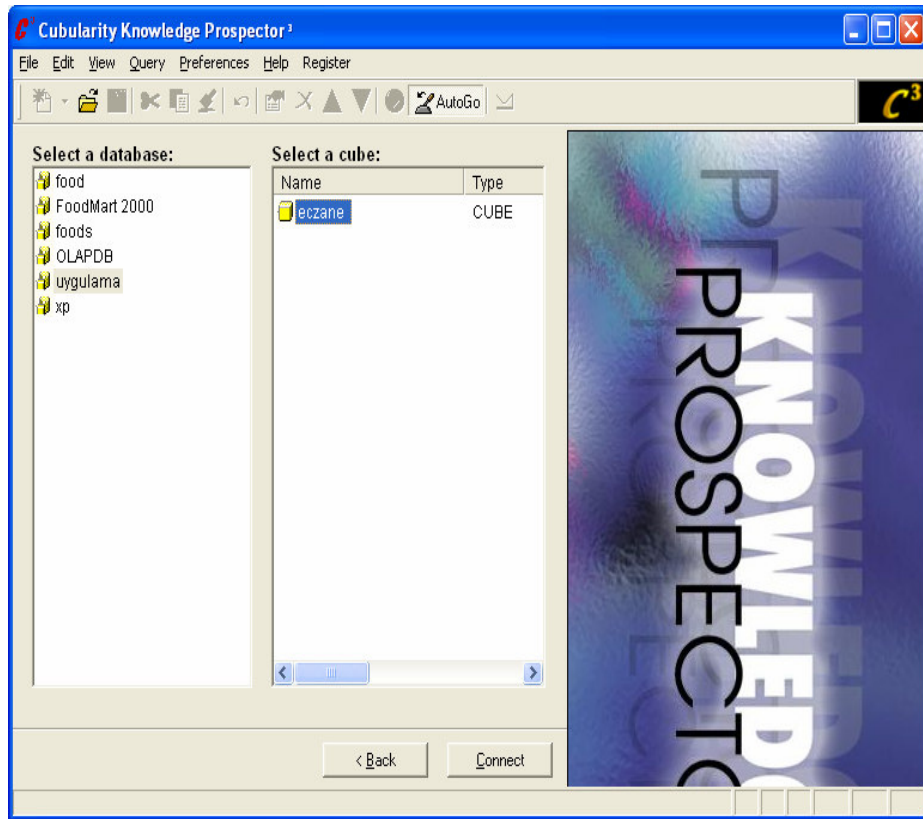
5.3.2. Knowledge Prospector ile analizlerin oluşturulması

Analiz Servisi'nde oluşturulan eczane küpü ile bağlantı Şekil 5.25.'de görülen mevcut OLAP veritabanları arasından seçim yapılarak kurulmaktadır.

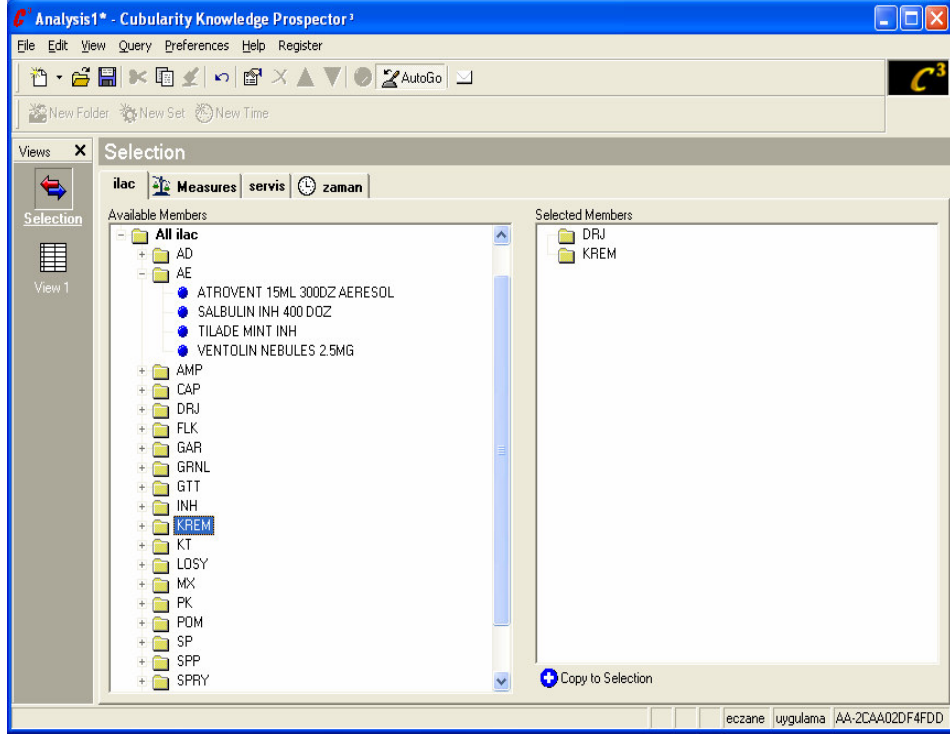
Oluşturulacak analizde eczane küpünde mevcut olan ilaç boyutundan krem ve draje türündeki ilaçların dahiliye ve acil servislerinde 2000 ve 2001 yıllarında kullanım miktarları ve bu servislerden elde edilen net gelirler incelenmek istenmektedir. Bunun için gereken seçimler Knowledge Prospector'da

seçim menüsünden Şekil 5.26.'da olduğu gibi sürükle bırak yöntemiyle yapılabilmektedir.

Analiz için gerekli boyut üyeleri ve ölçeklerin belirlenmesinden sonra verinin gösterimini gerçekleştirmek amacıyla seçim penceresinden görünüm penceresine geçilmektedir. Görünüm penceresinde analizin hangi tür gösterim biçimiyle gerçekleştirileceği seçilebilmektedir. Görünümler seçilen boyut ve ölçeklerin grafik alanına sürükle bırak yöntemiyle yerleştirilmesi sonucu hazırlanmaktadır. Şekil 5.27.'de yapılan analizin tablo biçiminde görüntüsü ve Şekil 5.28.'de grafiksel olarak hazırlanmış görüntüsü görülmektedir.



Şekil 5.25. OLAP küpü ile bağlantı kurulması



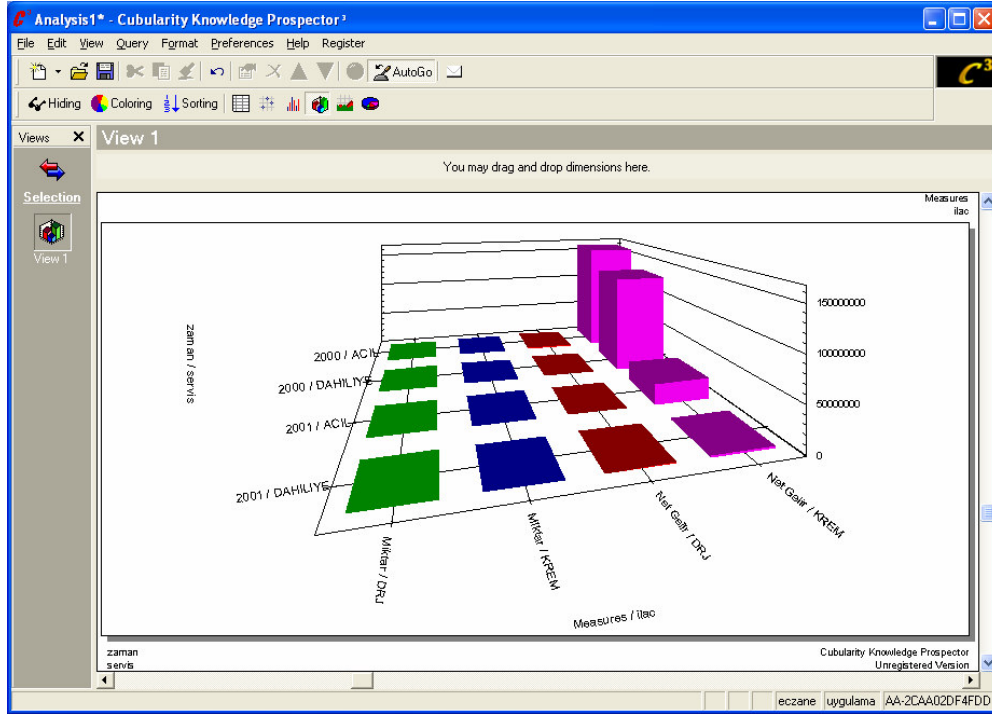
Şekil 5.26. Analiz edilecek üyelerin seçimi

The screenshot shows the 'View 1' view in the Analysis1* software. The table displays data for 'Miktar' and 'Net Gelir' across years 2000 and 2001, categorized by 'ACIL' and 'DAHILIYE'. The table structure is as follows:

		2000		2001	
		ACIL	DAHILIYE	ACIL	DAHILIYE
Miktar	DRJ	23,00	8,00	1,00	14,00
	KREM	29,00	28,00	7,00	6,00
Net Gelir	DRJ	1.526.763,00	354.032,00	22.127,00	570.878,00
	KREM	159.501.914,00	130.965.800,00	24.859.730,00	2.198.214,00

The status bar at the bottom indicates 'eczane uygulama AA-2CAA02DF4FDD'.

Şekil 5.27. Tablo biçiminde gösterim



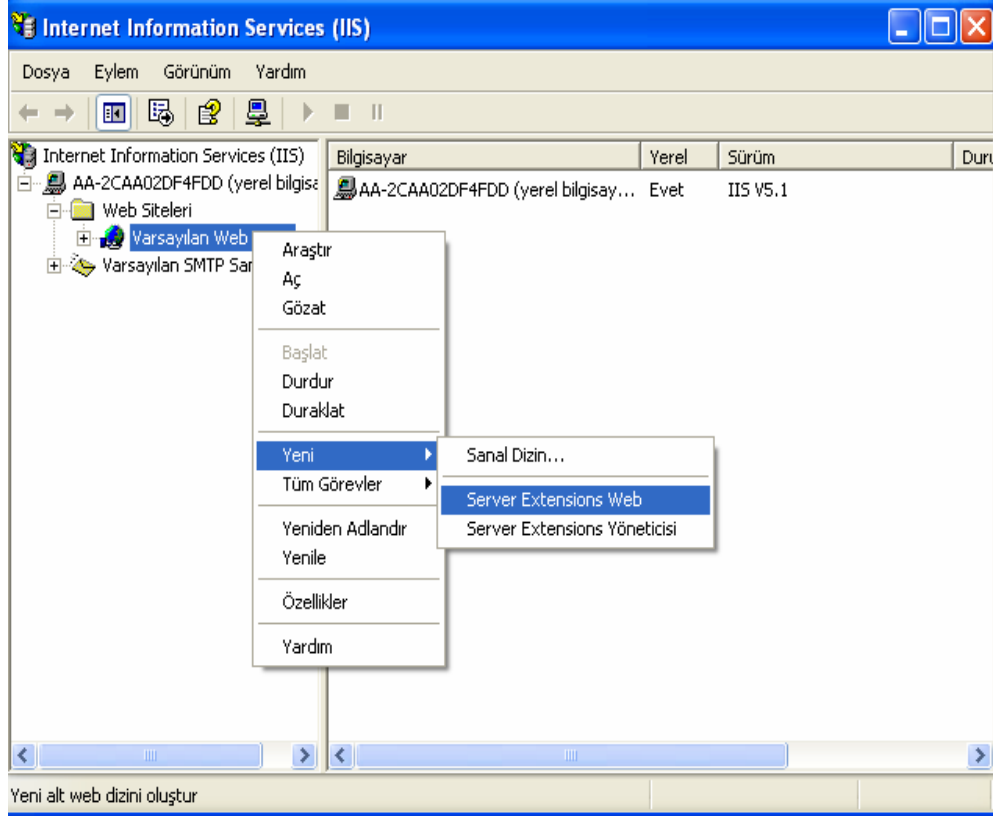
Şekil 5.28. Grafiksel gösterim

5.3.3. Web yoluyla erişimin sağlanması

Gerçekleştirilen analizlerin internet ya da intranet ortamında diğer kullanıcılar ile paylaşılması için Knowledge Server'dan yararlanılmaktadır. Bunun için Web Renderer kurulmalıdır.

5.3.3.1. Web Renderer kurulumu

Web Renderer kurulumu için kullanılacak makinede IIS 5.0 yüklü olması gerekmektedir. Öncelikle Web Renderer için sunucu olarak kullanılacak makinede Server Extensions Web seçeneği seçilerek alt web dizini oluşturulmalıdır. Şekil 5.29.'da alt web dizinin oluşturulması gösterilmektedir.



Őekil 5.29. Alt web dizininin oluřturulması

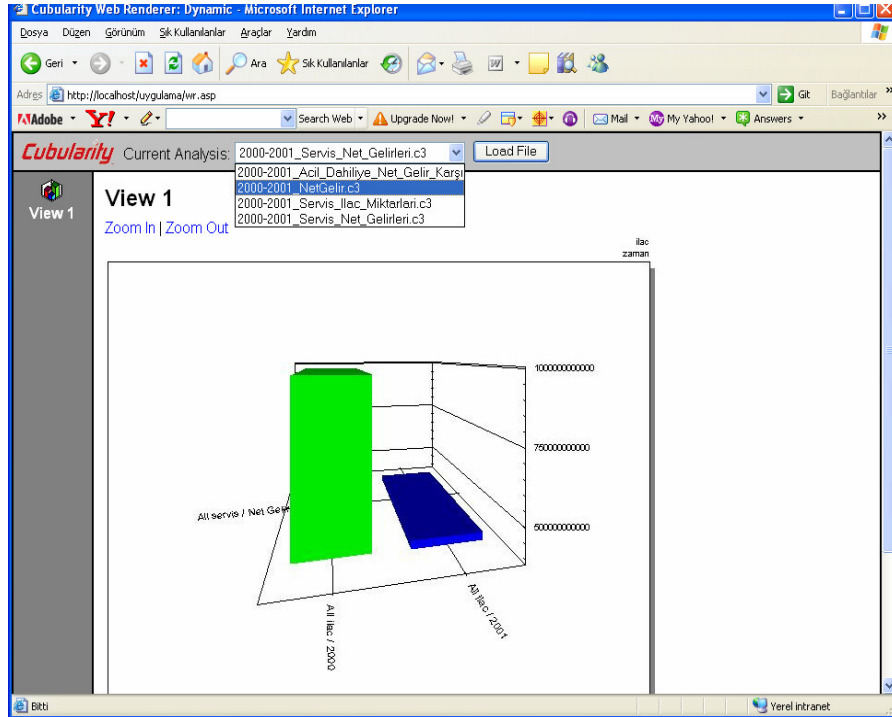
Oluřturulan alt dizine Web Renderer özelliğinin alıřtırılabilmesi iin Knowledge Server ile birlikte gelen ařağıdaki dosyaların yklenmesi gerekmektedir:

- default.asp: Web Renderer ile ilgili bilgileri saėlayan bir bařlangı sayfasıdır.
- wr.asp: Seilen analiz ve grnmleri dinamik olarak oluřturan ve bir web arayz saėlayan ASP sayfasıdır.
- wr_static.asp: Statik Web Renderer iin XML dosyalarını grntler ve web arayz saėlar.
- library.asp: Kaydedilmiř analizlere eriřim iin bir ktphane oluřturur.
- interface.asp: Tm Web Renderer sayfalarının tasarımı ve temelini oluřturmaktadır.
- wr.css: Web arayznn grnmn belirleyen dosyadır.

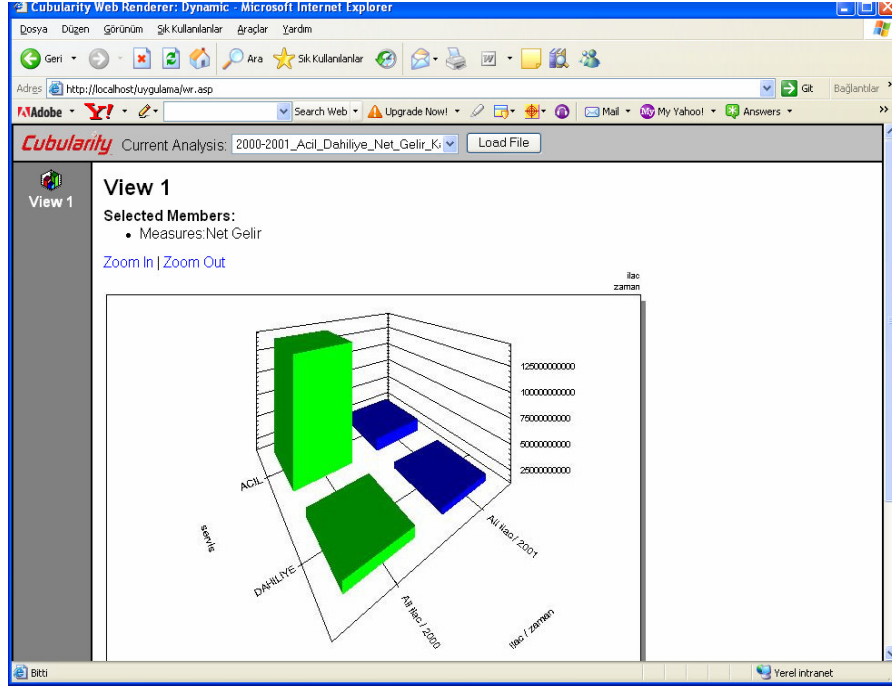
- global.asa: Bir kullanıcı için açılan oturumda oluşturulan imaj dosyalarının temizlenmesini gerçekleştirmektedir.
- register.asp: Knowledge Server kaydını gerçekleştirir.
- images: Web Renderer için gerekli imaj dosyalarını içinde barındıran klasördür.
- analyses: Web Renderer ile paylaşılan tüm analiz dosyalarını barındıran dizindir. Paylaşılacak analizler bu dizin içerisine kaydedilmelidir.

5.3.3.2. Oluşturulan analizlerin web tarayıcıda görüntülenmesi

Knowledge Prospector ile eczane küpünden yararlanılarak oluşturulan analizler analyses klasörüne kopyalanarak web tarayıcı ile kullanıcılar arasında paylaşılmaktadır. Web tarayıcıda açılan ASP sayfasından istenilen analiz seçilebilmekte ve yükle komutu ile görüntülenebilmektedir. Şekil 5.30. ve Şekil 5.31.'de Web tarayıcıda analiz örnekleri gösterilmektedir.



Şekil 5.30. Web tarayıcıda analizlerin görüntülenmesi



Şekil 5.31. 2000-2001_Acil_Dahiliye_Net_Gelir.c3 analiz dosyası

6. SONUÇ

İşletmelerin rekabetçi ortamdaki başarıyla sıyrılarak varlıklarını devam ettirebilmeleri için kararların gerçeklere dayanarak sağlıklı bir şekilde alınması gerekmektedir. Bu ise mevcut durumun, doğru ve güvenilir veri kaynakları kullanılarak farklı yönlerden analizinin yapılmasıyla mümkün olmaktadır. İşletmenin mevcut durumunu olabilecek en iyi şekilde yansıtabilecek ve karar vericilere alacakları taktiksel ve stratejik kararlarda destek sağlayacak analiz sistemlerine ihtiyaç duyulmaktadır. Günümüzde yaygın olarak kullanılan sistemler bu ihtiyaçlara yanıt verememektedir. İşletme verileri doğası gereği çok boyutlu yapıdadır. Zaman, bölge, ürün, müşteri gibi birçok boyuta ve değişkene bağlı olan işletme verilerinin çok boyutlu olarak analiz edilmesi gerekmektedir.

Veri ambarları çok boyutlu analiz için temel oluşturmaktadır. Veri ambarları kullanılarak veriler, operasyonel bilgiden karar destek bilgisi için temel oluşturulan verilere dönüştürülmektedir. Geçmişe oranla daha fazla veri yığınlarıyla karşılaşıldığı ve verilerin daha karmaşık bir yapıya sahip olduğu günümüzde, sağlıklı kararlar almak isteyen işletmeler için veri ambarları doğru ve güvenilir veri kaynakları olarak vazgeçilmez olmaktadır. Veri ambarları ile veriler çok boyutlu analize uygun hale getirilerek karar destek sürecinde kullanılacak verilere dönüştürülmektedir.

Ellerindeki verileri en iyi şekilde değerlendirmek isteyen yöneticilerin, bu verileri çeşitli açılardan görmeden doğru bir fikir elde edebilmeleri mümkün değildir. Yöneticiler için ellerinde işletmeyle ilgili verilerin bulunması bir anlam ifade etmemektedir. Önemli olan ellerinde bulunan verilerin bir anlam ifade eder hale gelmesidir. Bu da OLAP araçları gibi çok boyutlu analiz araçlarıyla mümkündür.

Bugün işletmelerin geleneksel raporlama araçları kullanılarak üretilen raporlardan istenilen verimi almaları söz konusu değildir. Karmaşıklaşan iş dünyası ve iş verileri daha karmaşık sorgular ile daha karmaşık raporlamaların gerçekleştirilmesini zorunlu kılmaktadır. Yöneticilerin bu zorunluluğun farkına vararak, mevcut sistemlerine OLAP araçlarını dahil etme yoluna gitmeleri gerekmektedir. Ayrıca bu araçların kullanımı için gerekli olan personelin eğitimi ve eğitim maliyetleri üzerinde önemle durulması gereken bir diğer konudur.

İřletmeler OLAP aralarının kullanımı konusunda kullanıcılara gerekli eđitimi sađlamalı ve eđitim maliyetlerini gz nnde bulundurmalıdır.

Analiz ve raporlama aracı reticileri aısından bakıldıđında ise bu reticilerin ok boyutlu analiz ihtiyaını karřılamak amacıyla mevcut rnlerine OLAP iřlevselliđini entegre etmeye bařladıkları grlmektedir.

OLAP aralarının web temelli olarak kullanılması ise bu araları kullanan iřletmelerin nemle zerinde durmalarını gerektiren bir seenektir. OLAP araları iin geleneksel istemci sunucu mimarisi yerine web temelli bir alt yapı hazırlamak hem daha ok sayıda kullanıcının karar verme srecine dahil edilmesini sađlayacak hem de maliyetleri dřrecektir. KDS konusunda yazılım reten yazılım firmaları ise istemci sunucu mimarilerini web temelli olarak geliřtirmeye alıřmaktadırlar.

nmzdeki yıllarda veri ambarları ve OLAP aralarının, birbirlerini tamamlayan teknolojiler olarak web teknolojileri ile birlikte iřletmelerin biliřim teknolojileri yatırımlarında n sıralara ıkması mmkn gzlmektedir.

KAYNAKLAR

- [1] Power, D.J., 2006 <http://www.dssresources.com/tour/index.html>
- [2] Bhatt, G.D. ve Zaveri, J., "The enabling role of Decision Support Systems in organizational learning", *Decision Support Systems*, **32**, 297-309, 2002.
- [3] Marakas, G.M., *Decision Support Systems in the 21st Century*, Prentice Hall, New Jersey, A.B.D., 2003.
- [4] Power, D.J., *Decision support systems : Concepts and resources for managers*, Quorum Books, Westport, A.B.D., 2002.
- [5] Shim, J.P., Warkentin, M., Courtney, J.F., Power, D.J., Sharda, R. ve Carlsson, C., "Past, present and future of decision support technology", *Decision Support Systems*, **33**, 111-126, 2002.
- [6] Vieira, M. ve Madeira, H.; "Benchmarking the dependability of different OLTP systems, Dependable systems and networks", *2003 International Conference on Dependable Systems and Networks*, The Printing House, A.B.D., 305-310, 2003.
- [7] Meyer, D. ve Cannon C., *Building a better datawarehouse*, Prentice Hall, New Jersey, A.B.D., 1998.
- [8] Kimball, R., *The data warehouse toolkit: practical techniques for building dimensional datawarehouses*, John Wiley&Sons Inc., New York, A.B.D., 1996.
- [9] Anonim, *Welcome to PowerOLAP: Concepts and Benefits*, 2006.
http://www.paristech.com/PowerOLAP_help_files/powerolap/1_introduction/welcome_to_powerolap_concepts_and_benefits.htm
- [10] Beynon, M., Rasmeyan, S. Ve Russ, S., "A new paradigm for computer-based decision support", *Decision Support Systems*, **33**, 127-142, 2002.
- [11] Stroman, J. ve Wilson, K., Wauson, J., *Administrative assistant's and secretary's handbook*, Amacom, New York, A.B.D., 2003.
- [12] Hasan, H. ve Hyland, P., "Using OLAP and multidimensional data for decision making", *IT Pro*, **3**, 44-50, 2001.
- [13] Codd, E.F., Codd, S.B. ve Salley, C.T., *Providing OLAP to user analysts: An IT mandate*, 1993.

http://dev.hyperion.com/resource_library/white_papers/providing_olap_to_user_analysts.pdf

- [14] Niemi, T., Hirvonen, L. ve Jarvelin, K., “Multidimensional data model and query language for informetrics”, *JASIST*, **54**, 939-951, 2003.
- [15] Shekhar, K., *Develop OLAP applications with oracle express objects, Oracle,2000. develop, volume 1, student guide*, Oracle Corp, 2000.
- [16] Singh, S.,K., Watson, H.J. ve Watson, R.T., “EIS support for the strategic management process”, *Decision Support Systems*, **33**, 71-85, 2002.
- [17] Inmon, W.H., *Building the data warehouse*, John Wiley&Sons Inc., New York, A.B.D., 2002.
- [18] Thomsen, E., *OLAP solutions: Building multidimensional information systems*, John Wiley&Sons Inc., New York, A.B.D., 2002.
- [19] Alter, S., *Decision support systems occurrent practice and continuing challenges*, Addison-Wesley, New York, A.B.D., 1999.
- [20] Baragoln, C., Berclanos, J., Kornel, J., Robinson, G., Sawa, R. ve Schulnder, E., *DB2 OLAP server theory and practices*, IBM Corp., California, A.B.D., 2001.
- [21] Pendse, N., *What is OLAP?*, 2005. <http://www.olapreport.com/fasmi.htm>
- [22] Oracle Corp., *Oracle9i OLAP User’s Guide, Release 2 (9.2.0.2)*, 2002.
- [23] Malinowski, E. ve Zimanyi, E., “Hierarchies in a multidimensional model:From conceptual modeling to logical representation”, *Data&Knowledge Engineering*, 2005.
- [24] Lechtenbörger, J. ve Vossen, G., “Multidimensional normal forms for data warehouse design”, *Information Systems*, **28**, 415-434, 2003.
- [25] Shapiro, J., *SQL server 2000: The complete reference*, McGraw-Hill Professional, A.B.D., 2000.
- [26] Inmon, W.H., *Building the data warehouse*, John Wiley&Sons Inc., New York, A.B.D., 2002.
- [27] Singh, H.S., *Data warehousing: Concepts, technologies, implementations, and management*, Prentice Hall, New Jersey, A.B.D., 1998.
- [28] Han, J. ve Kamber, M., *Data mining: Concepts and techniques*, 2006. <http://www.cs.wmich.edu/~yang/teach/cs595/han/ch2.pdf>

- [29] Anonim, *Red Brick Corporate Background*, 2006.
<http://www.prnewswire.com/cnoc/CORPORATES/105756.5>
- [30] Sperley, E., *The enterprise datawarehouse: Planning, building, and implementation*, Prentice Hall, New Jersey, A.B.D., 1999.
- [31] Lehn, R., Lambert, V. ve Nachouki, M.P., “Data warehousing tool's architecture: from multidimensional analysis to data mining”, *Eighth International Workshop on Database and Expert Systems*, Toulouse, Fransa, 636-643,1997.
- [32] Koutsoukis, N.S., Mitra, G. ve Lucas, C., “Adapting online analytical processing for decision modelling: the interaction of information and decision technologies”, *Decision Support Systems*, **26**, 1-30, 1999.
- [33] Bain, T., Benkovich, M., Dewson, R., Ferguson, S., Graves, C., Joubert, T.J., Lee, D., Scott, M., Skoglund, R., Turley, P. ve Youness, S., *Professional SQL server 2000 data warehousing with analyses services*, Wrox Press, Chicago, A.B.D., 2001.
- [34] Anonim, *The power of OLAP*, 2006.
<http://survey.pearsonnncs.com/analysis/olap.htm>
- [35] Anonim, *DB2 Business Intelligence, Hierarchies*, 2006.
http://publib.boulder.ibm.com/infocenter/db2luw/v8/index.jsp?topic=/com.ibm.db2.udb.db2_olap.doc/cmdhierarchy.htm
- [36] Ponniah, P., *Data warehousing fundamentals: A comprehensive guide for IT professionals*, John Wiley&Sons Inc., New York, A.B.D., 2001.
- [37] Gray, P. ve Watson, H.J., *Decision support in the data warehouse*, Prentice Hall, New Jersey, A.B.D., 1998.
- [38] Bischoff, J. ve Alexander, T., *Data warehouse: Practical advice from experts*, Prentice Hall, New Jersey, A.B.D., 1997.
- [39] Dramowicz, K., *Creating and manipulating multidimensional tables with locational data using OLAP cubes*, 2005.
http://www.directionsmag.com/article.php?article_id=733&trv=1
- [40] Datta, A. ve Thomas, H., “The cube data model: A conceptual model and algebra for on-line analytical processing in data warehouses”, *Decision Support Systems*, **27**, 289-301, 1999.

- [41] Nepomnjashiy, A., *OLAP and Data warehousing: The problem and solution*, 2002.
<http://www.sqlservercentral.com/columnists/ANepomnjashiy/olaproblemandsolution.asp>
- [42] Schwenk, H. ve Hope, M., *The anatomy of an OLAP tool*, Ovum Ltd., 2002.
- [43] Sowmyanarayan, M. ve Munsh, A., *OLAP strategies for large data warehousing projects*, 2004.
<http://www.datawarehouse.com/article/?articleid=4697>
- [44] Potgieter, J., *OLAP data scalability*, 2003.
http://www.dmreview.com/article_sub.cfm?articleId=7636
- [45] Anonim, *MOLAP, ROLAP and HOLAP*, 2006.
<http://www.1keydata.com/datawarehousing/molap-rolap.html>
- [46] Anonim, *HOLAP*, 2006. <http://www.auditmypc.com/acronym/HOLAP.asp>
- [47] MicroStrategy Inc., *Scalable enterprise OLAP requirements: Data depth and breadth*, White Paper, 1998.
<http://www.microstrategy.com/Publications/Whitepapers/>
- [48] Howson, C., *BI Scorecard: OLAP*, 2004.
http://www.intelligententerprise.com/print_article.jhtml?articleID=19502190
- [49] Eom, S.B., *Inter-organizational information systems in the internet age*, Idea Group, Pennsylvania, A.B.D., 2004.
- [50] Targowski, A., *Electronic enterprise: Strategy and architecture*, Idea Group, Pennsylvania, A.B.D., 2003.
- [51] Anonim, *XML*, 2006. http://www.idealliance.org/standards_xml.asp
- [52] Anonim, *ASP Tutorials*, 2006.
http://www.webwizguide.info/asp/tutorials/what_is_esp.asp
- [53] Poole, J., "WOLAP Packs A Wallop - New tools make Web-based OLAP a sure bet", *VARbusiness*; 1402, **42**, 1998.
- [54] Taylor, A., *The warehouse meets the web*, 1999.
<http://www.tdan.com/i004fe04.htm>

- [55] Carickhoff, R., *A new face for OLAP*, Internet Systems, 1997.
<http://www.dbmsmag.com/9701i08.html#table1>
- [56] Runnels, K., *Special Edition Using Microsoft BackOffice, 2*, 2006.
<http://www.is-edu.hcmuns.edu.vn/WebLib/Books/Database/0-7897-1130-3/ch37/ch37.htm>
- [57] Richmond-Coggon, M. ve Wolf, K., *Delivering the power of OLAP around the globe*, 2006.
http://www.revelwood.com/downloads/delivering_the_power_of_olap.pdf
- [58] *Cubularity Knowledge Platform Data Analysis and Reporting for the Entire Enterprise*, 2006
www.cubularity.com