

**VERİ AMBARI VE OLAP TEKNOLOJİLERİNDEN
YARARLANILARAK RAPORLAMA ARACI GERÇEKLEŐTİRİMİ**

**Pamukkale Üniversitesi
Fen Bilimleri Enstitüsü
Yüksek Lisans Tezi
Bilgisayar Mühendisliđi Anabilim Dalı**

Devrim İŐLİ

Danışman: Yrd. Doç. Dr. Gürhan GÜNDÜZ

**Eylül 2009
DENİZLİ**

YÜKSEK LİSANS TEZİ ONAY FORMU

Devrim İŞLİ tarafından Yrd. Doç. Dr. Gürhan GÜNDÜZ yönetiminde hazırlanan “Veri Ambarı ve OLAP Teknolojilerinden Yararlanılarak Raporlama Aracı Gerçekleştirimi” başlıklı tez tarafımızdan okunmuş, kapsamı ve niteliği açısından bir Yüksek Lisans Tezi olarak kabul edilmiştir.


Yrd. Doç. Dr. Gürhan GÜNDÜZ
Süri Başkanı (Danışman)


Yrd. Doç. Dr. Kadir KAVAKLIOĞLU
Jüri Üyesi


Yrd. Doç. Dr. A.Kadir YALDIR
Jüri Üyesi

Pamukkale Üniversitesi Fen Bilimleri Enstitüsü Yönetim Kurulu'nun ~~30.09/2009~~ tarih ve ~~21/25~~ sayılı kararıyla onaylanmıştır.


Prof. Dr. Halil KARAHAN
Müdür

Bu tezin tasarımı, hazırlanması, yürütülmesi, araştırılmalarının yapılması ve bulgularının analizlerinde bilimsel etiğe ve akademik kurallara özenle riayet edildiğini; bu çalışmanın doğrudan birincil ürünü olmayan bulguların, verilerin ve materyallerin bilimsel etiğe uygun olarak kaynak gösterildiğini ve alıntı yapılan çalışmalara atfedildiğini beyan ederim.

İmza:

Öğrenci Adı Soyadı: Devrim İŞLİ

TEŐEKKÖR

Danışman hocam saygı deęer Yrd. Doę. Dr. Gürhan GÖNDÖZ'e proje alıřmam boyunca bana her konuda yardımcı olup anlayıř göstermesinden dolayı teőekkürlerimi sunarım.

ÖZET

VERİ AMBARI VE OLAP TEKNOLOJİLERİNDEN YARARLANILARAK RAPORLAMA ARACI GERÇEKLEŞTİRİMİ

İŞLİ, Devrim
Yüksek Lisans Tezi, Bilgisayar Mühendisliği ABD
Tez Yöneticisi: Yrd. Doç. Dr. Gürhan GÜNDÜZ
Eylül 2009, 82 Sayfa

Günümüzde pek çok organizasyon kendi bünyesindeki günlük işlemleri takip etmek amacıyla hazırlanmış otomasyon yazılımlarını kullanmaktadır. Bu yazılımlar kayıtlarını yüksek seviyeli hareket işleme amacıyla tasarlanmış ilişkisel veritabanlarında saklarlar. Zaman içerisinde işletmelerin sistemlerindeki veri boyutları hızlı bir şekilde artar ve farklı isteklerin de etkisiyle veri bozulmaları meydana gelir. Sistem içerisindeki bu hızlı büyüme ve veri bozulmaları verilerin anlaşılması ve analizinde problemlerin ortaya çıkmasına sebep olur. Bu zorluklar göz önünde bulundurularak farklı uygulamalarla ilgili kayıtları ortak bir çatı altında toplamak, sorgulamaya uygun bir şekilde bir arada tutmak amacıyla veri ambarı platformundan faydalanır. Bu sayede güncel veriyi tutan ve yoğun veri girişi yapılan OLTP (Online Transaction Processing) sistemlerinden çeşitli raporlar alarak bu sistemlerin performansını kötü yönde etkilememiş olur. Veri ambarı içindeki veri belirli işlemler ve hesaplamalar sonrası hazırlanır. Bu hesaplamalar sorgu sırasında artı bir performans getirisi sağlamaktadır.

Çok boyutlu analizlerin yapılma ihtiyacının karşılanması için kullanılan OLAP (Online Analytical Processing) teknolojisi ile ilişkisel veritabanlarına göre önemli performans kazanımları elde edilebilmektedir.

Bu çalışmanın konusu Pamukkale Üniversitesi Eğitim Uygulama ve Araştırma Hastanesi'nde kullanılan HBYS (Hastane Bilgi Yönetim Sistemi) ve MYS (Malzeme Yönetim Sistemi) ilişkisel veri tabanları kullanılarak bir veri ambarı tasarlanıp, geliştirilmesidir. Öncelikle veri ambarı ve veri ambarı işlemlerinin adımları tanımlanmakta, ardından uygulama kısmı anlatılmaktadır.

Veri ambarının oluşturulmasının ana amacı çekilen anlık raporlamaların karar destek mekanizmasını oluşturan birçok verinin günlük aylık, yıllık olarak servisler, doktorlar, bölümler, kişiler bazında incelenmesi sağlamaktır. Doktor bazında muayene sayısı ve gelir, müdahale sayısı ve birimleri, protokol sayıları, kurum borç alacak durumları, doktorların tetkik istem sayıları ve ayrıntıları, eski/yeni hasta oranları ve

birçok istatistikî bilgi veri ambarında toplanıp, tarih bazında karşılaştırmalı olarak grafiklerle sunulabilir.

Anahtar Kelimeler: Veri Ambarı, OLAP

Yrd. Doç. Dr. A.Kadir YALDIR

Yrd. Doç. Dr. Gürhan GÜNDÜZ

Yrd. Doç. Dr. Kadir KAVAKLIOĞLU

ABSTRACT

IMPLEMENTATION OF A REPORT TOOL WITH DATA WAREHOUSE AND OLAP TECHNOLOGIES

İŞLİ, Devrim
M. Sc. Thesis in Computer Engineering
Supervisor: Asst. Prof. Dr. Gürhan GÜNDÜZ
September 2009, 82 Pages

Nowadays many organisations have been using automation software that are prepared to follow the daily processes within the organisation. These software store their records in databases that are designed for high level transactions. In time, data size inside the systems of the organisations grows rapidly and with the effect of different requests data corruptions occurs. These rapid growth and data corruptions inside the system cause problems in understanding and analysing the data. Regarding these difficulties, we benefit from datawarehouse to collect the records related to different applications under same cover and to keep them together for appropriate questioning, consequently it won't affect the performance of systems badly by getting some reports from OLTP (Online Transaction Processing) systems that keep the up-to-date data and get intensive data input. The data inside the data warehouse are prepared after certain transactions and calculations. These calculations provide a positive performance gain during questioning.

With OLAP(Online Analytical Processing) technology that is used to meet the demand to use multi dimensional analyses important performance profits are possible compared with relational database.

Topic of this work has been improved by desinging a data warehouse by using the relational databases: HBYS(Hospital's Information Managing System) and MYS (Material Managing System) which are being used in Pamukkale Üniversitesi Eğitim Uygulama ve Araştırma Hastahanesi (Pamukkale University Education Application and Research Hospital). First, datawarehouse and steps of processes of datawarehouse are defined and then application part is explained.

The main purpose to form a datawarehouse is to enable processing lots of data that form decision support mechanism of time consuming reportings; by services, doctors, departments, and people daily, monthly, and yearly. For doctors number of examination

and revenue, number of intervention and units, protocol numbers, institution's debt-will situations, number of doctors' technical request and details, previous/current patient rates and lots of statistical information are able to be brought in with graphics dately comparative by storing in dataware.

Keywords: Data Warehouse, OLAP

Asst. Prof. Dr. A.Kadir YALDIR

Asst. Prof. Dr. Gürhan GÜNDÜZ

Asst. Prof. Dr. Kadir KAVAKLIOĞLU

Sayfa No

İçindekiler	viii
Şekiller Dizini	x
Simge ve Kısaltmalar Dizini	xii
1. GİRİŞ	1
2. VERİ AMBARI	3
2.1 Giriş	3
2.2 Veri Ambarı ile OLTP Sistemleri Arasındaki Farklar	5
2.3 Veri Ambarı Mimarisi	9
2.3.1 Kurumsal Veri Ambarı	9
2.3.2 Veri Pazarı (Data Mart)	9
2.3.3 Sanal Veri Ambarı	9
2.4 Veri Ambarı Tasarlanması	11
2.5 Veri Ambarı Uygulama Örnekleri	13
3. OLAP SİSTEMİ	16
3.1 Giriş	16
3.2 OLAP Sunucu Türleri	19
3.2.1 ROLAP	19
3.2.2 MOLAP	19
3.2.3 HOLAP	20
3.3 OLAP Mimarisi	21
3.3.2 Olaylar	23
3.3.3 Boyutlar	23
3.3.4 Ölçüm Değerleri	24
3.4 OLAP Küplerinde Genel İşlemler	25
3.5 OLAP Tasarımı	28
3.5.1 Yıldız Şema (Star Schema)	28
3.5.2 Kar Tanesi Şema (Snowflake Schema)	29
3.6 OLAP Sorgulama Yöntemleri	30
3.6.1 MDX	31
4. TEZ KAPSAMINDA KULLANILAN ÜRÜNLER	32
4.1 Giriş	32
4.2 Microsoft Visual Studio .NET 2008	33
4.3 Microsoft Sql Server 2008	33
4.4 Microsoft Analysis Services 2008	33
4.5 Microsoft Reporting Services 2008	34
5. VERİ AMBARI VE OLAP TEKNOLOJİLERİNDEN YARARLANILARAK RAPORLAMA ARACI GERÇEKLEŞTİRİMİ	36
5.1 Hastane Bilgi Yönetim Sistemi Yapısı	37
5.2 Veri Ambarı Tasarımı	39
5.2.1 Veri Ambarı Olay (Fact) Tabloları	39
5.2.2 Veri Ambarı Boyut (Dimension) Tabloları	40
5.3 Veri Ambarı Üzerinde Küp ve Boyutların Tanımlanması	44
5.4 Veri Ambarını Dolduracak Verilerin Seçilmesi ve Temizlenmesi	53
5.5 Karar Destek Amaçlı Sorgulama ve Raporlama Aracı Gerçekleştirimi	56
5.5.2 MDX Sorgulama	61

5.6	Veri Ambarı ve Veri Tabanı Karşılaştırılması	64
6.	SONUÇLAR VE DEĞERLENDİRME	67

ŞEKİLLER DİZİNİ

	<u>Sayfa No</u>
Şekil 2.1 OLTP Sistemlerdeki İşlemler	6

Şekil 2.2 Veri Ambarı Sistemindeki İşlemler	7
Şekil 2.3 Veri Ambarı Mimarisi	10
Şekil 2.4 Veri Ambarı Bileşenleri	11
Şekil 3.1 Veritabanı ve Veri Ambarı ilişkisi ve Bilgi Mimarisi	17
Şekil 3.2 OLAP Mimarisi	18
Şekil 3.3 ROLAP Mimarisi	19
Şekil 3.4 MOLAP Mimarisi	20
Şekil 3.5 HOLAP Mimarisi	21
Şekil 3.6 Küp Yapısı	22
Şekil 3.7 Üç Boyutlu Küboid Hiyerarşik Genel Kavramı	23
Şekil 3.8 Olay Tablo Yapısı	24
Şekil 3.9 Küp İçerisinde Veriye Erişim	25
Şekil 3.10 Roll Up İşlemi	26
Şekil 3.11 Drill Down İşlemi	27
Şekil 3.12 OLAP Küpleri Genel İşlemleri	27
Şekil 3.13 Yıldız Şema	29
Şekil 3.14 Kar tanesi Şema	30
Şekil 4.1 Analysis Services	34
Şekil 4.2 Reporting Services	35
Şekil 5.1 Acil İşlemler ER Diagramı	42
Şekil 5.2 Hasta Hizmetleri ER Diagramı	42
Şekil 5.3 Birime Çıkılan Malzemeler ER Diagramı	43
Şekil 5.4 Ameliyat İşlemleri ER Diagramı	43
Şekil 5.5 Laboratuar İşlemleri ER Diagramı	44
Şekil 5.6 Analiz Sunucuda Yeni Veri Kaynağı Tanımlanması	46
Şekil 5.7 Analiz Sunucuda Yeni Veri Kaynağı Tanımlanması	47
Şekil 5.8 Küp oluşturma ve Ölçüm Tablo Seçimi	48
Şekil 5.9 Küp Ölçüm Değerlerinin Belirlenmesi	48
Şekil 5.10 Küp Boyut Değerlerinin Belirlenmesi	49
Şekil 5.11 Dimension Usage	49
Şekil 5.12 Calculations	50
Şekil 5.13 Calculations Script Örneği	51
Şekil 5.14 Küp Yapısının Düzenlenmesi	52
Şekil 5.15 Küp Yapısının Analiz Sunucuya Yüklenme İşlemi	53
Şekil 5.16 “Hastaya Çıkılan Malzemeler” olay tablosunun yüklenmesinde kullanılan SQL dili ile hazırlanmış görev içeriği	55
Şekil 5.17 Reporting Services Ekranı	56
Şekil 5.18 Reporting Services Ekranından Analysis Services’e Bağlanma	57
Şekil 5.19 Raporlama için OLAP Küp Seçimi	58
Şekil 5.20 “Birime Çıkılan Malzemeler” kübü kullanılarak oluşturulan MDX sorgulaması	58
Şekil 5.21 Birime Çıkılan Malzemeler Rapor Örneği	59
Şekil 5.22 Birime Çıkılan Malzemeler Aylara Göre Satış Sayıları	60
Şekil 5.23 Birime Çıkılan Malzemeler Düzey Adlarına Göre Satış Sayıları	60
Şekil 5.24 Birime Çıkılan Malzemeler Düzey Adlarına Göre Toplam Satış Sayıları	61
Şekil 5.25 “.Members” Operatörü ile seçilmiş bütün üyeler	63
Şekil 5.26 Veri Ambarı ve Veri Tabanı Karşılaştırması	65
Şekil 5.27 Sorgulamada Gelen Veri Miktarları	65
Şekil 5.28 Veri Ambarı ve Veri Tabanı Sorgulama Süreleri	66

SİMGE VE KISALTMALAR DİZİNİ

ÇBI	Çok Boyutlu İfade
ÇİAİ	Çevrimiçi Analitik İşleme

ÇİHI	Çevrimiçi Hareket İşleme
Data Warehouse	Veri Ambarı
Data Mart	Veri Pazarı
Data Mining	Veri Madenciliği
Dimension	Boyut
Dimension Table	Boyut Tablosu
DSS	Decision Support System
DTS	Data Transformation Services
ETL	Extract Transform Load
Fact Table	Olay Tablosu
HOLAP	Hybrid OLAP
HBYS	Hastane Bilgi Yönetim Sistemleri
KDS	Karar Destek Sistemleri
Measure	Ölçüm Değeri
MYS	Malzeme Yönetim Sistemi
MDX	Multi Dimensional Expression
MOLAP	Multidimensional OLAP
OLAP	Online Analytical Processing
OLTP	On-Line Transaction Processing
PTS	Pivot Table Service
ROLAP	Relational OLAP
SQL	Structured Query Language

1. GİRİŞ

İşletme ve kurumlar bilgi teknolojilerindeki gelişmelerin sağladığı olanaklar sayesinde çeşitli kaynaklarda üretilen verileri hızlı biçimde toplamakta ve bilgisayar ortamında depolamaktadır. Verileri saklama ve işleme ortamı olarak genelde veri tabanı kullanılmaktadır. OLTP (Online Transaction Processing, Çevrim İçi Hareket İşleme) sistemler günümüzde yaygın olarak kullanılan veri tabanı sistemlerine verilen isimdir. Veri tabanları, aynı anda birden fazla kullanıcının, verileri hızlı ve etkin bir şekilde girişi, çıkışı ve güncellenmesi için tasarlanmış yapılardır. OLTP veri tabanları; sürekli yeni veriler eklenen, verilerin modifiyesinin yapıldığı anlık durumlar için oluşturulurlar.

Verilerin bilgisayar ortamında saklanmasıyla birlikte sürekli artan veri miktarının saklanması için kullanılan veritabanları da aynı hızla artmıştır. Bu yapılar üzerinde hem veri analizi yapmak hem analiz yapacak algoritmaları çalıştırmak neredeyse imkânsızdır. Veritabanlarındaki saklı duran bilgilere ulaşmak genellikle çok zordur. Verideki aşırı büyüme istenilen bilgiye giderek daha zor ulaşmamıza neden olur. OLTP sistemlerden yapılan sorgular karmaşıklaştıkça hem sorgu performansını hem de sistemde çalışan diğer işlemleri (veri giriş, güncelleme ve silme) olumsuz etkiler. Bu nedenle veri tabanlarında saklanan veri, karar destek sistemlerinde kullanılabilecek türde bir veri değildir.

Zaman içerisinde veri büyüklüğünün artması ve analiz işlemlerinin zorlaşması nedeniyle OLTP sistemler ve sonrasında veri ambarı kullanımı artmıştır. Veri ambarında tutulan verilere hızlı erişim ve çok boyutlu analiz ihtiyacını karşılamak üzere OLAP (Online Analytical Processing, Çevrimiçi Analitik İşleme) teknolojisi gelişmiştir. OLAP teknolojisi ile güncel verileri tutan ilişkisel veritabanındaki bilgiler OLAP küplerine aktarılır ve buradan hızlı analizler yapılması sağlanır.

Belirli zaman aralıklarında OLTP sistemlerden, karar destek faaliyetlerinde kullanılacak özet verilerin, performans geliştirici çeşitli dönüşüm ve hesaplamalar sonrası veri ambarına aktarımı yapılır. OLTP sistemlerden veri ambarlarına veri transferi yapmak amacıyla ETL (Extract Transform Load) araçları bulunmaktadır. ETL işlemi, farklı kaynaklarda bulunan veriler çekilerek, mantıksal hatalar içeren kayıtların temizlenmesi, gerekli dönüştürmeler ve özetlemeler yapılarak veri ambarına yüklenmesidir.

Veri ambarı tasarımında denormalizasyon işlemlerinin kullanılması OLAP sorgularını işleme performansı nedeniyle tercih edilmektedir. OLAP veri modelleri performans işlemlerinin dışında verilere çok boyutlu bakma imkânı sağlar. OLTP veri modeli her OLTP uygulaması için ona uygun olarak tasarlanırken, OLAP veri modeli bütün OLTP uygulamalarının çapraz sorgulanmasına olanak sağlayacak bir yapıda tasarlanmalıdır. Veri ambarı içerisinde birden fazla OLTP kaynak sisteminde üretilen veriler bir araya getirilerek bir arada analiz edilebilir.

Bu proje kapsamında Pamukkale Üniversitesi Eğitim Uygulama ve Araştırma Hastanesi'nde kullanılan HBYS ve MYS ilişkisel veritabanları için gerçekleştirilen veri ambarı ve OLAP uygulaması tanıtılmaktadır. Proje kapsamında yapılan uygulama için yüksek veri hacmine teşkil eden, analiz ve raporlama ihtiyacının yüksek olduğu bir sektör olan sağlık sektöründe kullanılan veritabanları seçilmiştir.

Bölüm 2'de veri ambarı teknolojisi, terminolojisi, yapılandırma yöntemleri anlatılmakta ve uygulama örnekleri incelenmektedir.

Bölüm 3'de OLAP teknolojisini oluşturan bileşenler anlatılmaktadır.

Bölüm 4'te SQL Server 2008 içindeki OLAP bileşenleri incelenmektedir.

Bölüm 5'de proje içinde yapılan uygulama tanıtılmaktadır.

Bölüm 6'da sonuçlar ve değerlendirme verilmektedir.

2. VERİ AMBARI

2.1 Giriş

Son yıllarda teknolojiye yaşanan büyük gelişmeler sayesinde işletmelerde depolanan veriler çok büyük boyutlara ulaşmaya başlamıştır. Bu işletmelerde karışık organizasyon yapıları ve farklı veriler farklı yerleşim birimlerine dağılmış durumda olabilir. Karar Destek Sistemlerinin beslenmesi gereken girdilerin en önemlisi olan verilerin artmasıyla ve ilişkisel veritabanlarından ayrılma ihtiyacı doğunca veri ambarları gündeme gelmiştir.

Veri ambarı, bir işletmenin ya da kurumun çeşitli birimleri tarafından canlı sistemler aracılığı ile toplanan verilerin, ileride değerlendirmeye alınabilecek olanlarının geri planda yer alan bir sistemde birleştirilmesinden oluşan büyük ölçekli bir veri deposudur (Tantuğ, 2002). Yapılma amacı doğrultusunda kullanılmadığı sürece kendi başına değeri olmayan büyük miktarda verilerin barındırılmasını sağlayan veri ambarı ilişkili verilerin sorgulanabilindiği ve analizlerinin yapılabildiği geleneksel yöntemlerin yanında veri madenciliği ve çok boyutlu çözümleme (Multi Dimensional Analysis) gibi güçlü veri çözümleme teknikleri için altyapı sağlar.

Veritabanı sistemlerinin gelişimi üç ayrı fonksiyon ile değerlendirilmektedir:

- Veri toplama ve veritabanı oluşturma,
- Veri yönetimi (veri depolama, bilgi keşfi),
- Veri çözümlemesi ve veriyi anlama (veri ambarcılığı ve veri madenciliği gerektirir) (Han and Kamber, 2001).

Veri ambarı ile veritabanı temelde içerik bakımından birbirinden ayrılır. Geleneksel veritabanı sistemleri, kullanıcı hareketlerine bağlı günlük işlemleri desteklemek için tasarlanmıştır ve bu sistemler işletimsel (operational) ya da hareketsel (transactional)

sistemler olarak adlandırılır. Veritabanı yönetim sistemi olarak da adlandırılan bir veritabanı sistemi, veritabanı olarak bilinen birbiriyle ilgili verilerden oluşmaktadır. İlişkisel veritabanları, tablolardan oluşmaktadır. Veritabanları ham veriyi barındıran sistemlerdir. Ham veri yapılan işin anlık durumunu gösteren gerçeklerden oluşur.

Veri ambarları ise ham veriden türetilmiş veriyi barındırırlar. Türetilmiş veri, çeşitli bakış açılarından ve farklı içeriklerle ham veriden elde edilir ve birbiri ile ilişkili bir yapıdır. İşin amaç ve hedefine uygun bir yapıda tasarlanırlar.

W. H. Inmon'a göre; Veri ambarı, özneye dayalı, bütünleşmiş, zaman dilimli ve yöneticinin karar verme işleminde yardımcı olacak biçimde toplanmış olan değişmeyen veriler topluluğudur. Başlıca bu dört özellik veri ambarlarını, ilişkisel ve hareket veritabanı sistemlerinden ayıran özelliktir (Han and Kamber, 2001). Veri ambarında yer alan veri özelliklerini inceleyecek olursak:

- Özneye Dayalı: Veri ambarı bir organizasyonun her güne ait işleri ve hareket işleme faaliyetleri üzerinde yoğunlaşmak yerine, karar verecek kişiler için veriye ait modelleme ve çözümleme üzerinde yoğunlaşır. Öznenen yola çıkarak kullanılmayacak veriyi hariç tutar ve verinin basitleştirilmesini sağlar.
- Bütünleşik: Veri ambarı genellikle; birden çok ve farklı türdeki ilişkisel veritabanları ve farklı türdeki veri kaynağından alınan kayıtları bütünleştirerek oluşturulur. Veri temizleme ve bütünleme yöntemleri; isimlendirmede, şifreleme yapılarında, nitelik ölçütlerinde ve benzeri konularda tutarlılığı garantilemek için uygulanır. Bu işlemler ETL araçları kullanılarak yapılır.
- Zaman Dilimli: Veri ambarı içerisindeki her bir veri, tarihi bir bakış açısından bilgi sağlamak için depolanır (örneğin 5-10 yıllık geçmiş içerisinden).
- Değişmeyen: Veri ambarı, işletimsel (operational) sistemler farklı olarak hareket işleme (transaction), geri alma (rollback) ve aynı zamanda işleme (concurrent processing) gibi işlemleri içermez. Veriye erişim için sadece iki işlem gerektirir, verinin ilk yüklemesi ve verinin erişimi.

İşletme içinde birden fazla veritabanı uygulaması kullanımı olduğu durumlarda veri ambarı kullanımı daha yaygındır. Bunun nedeni ise heterojen kaynaklar kullanan ve veriye kolay erişim sağlayan bir yapının henüz sağlanamamasıdır. Bazı firmaların bu tür örnek

teşkil edebilecek ürünleri olsa da sundukları çözümlerin maliyeti çok yüksek olmaktadır. Bu ürünler temelde farklı tipte bir veritabanına aynı şekilde erişim sağlanabilmesi için bir aracı rolünü üstlenirler.

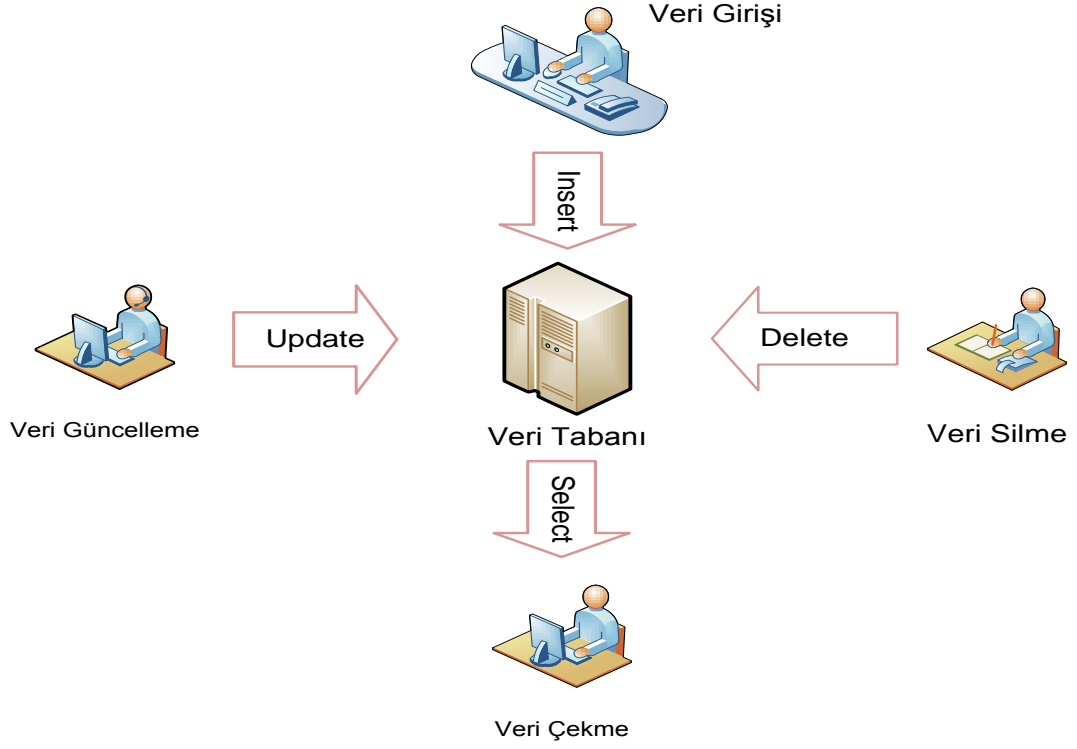
Veri ambarlama bu ürünleri kullanarak veriye anında erişme yöntemine alternatif olmaktadır. Herhangi bir sorgunun ilgili veritabanına iletilip sonucun işlenmesi yerine verinin tüm veri kaynaklarından önceden alınarak, dönüşümlerinin ve bütünleştirmelerinin yapılarak, hazır tutulan güncel bir kaynağın sorgulanması yaklaşımı benimsenmiştir. Verinin çeşitli aşamalardan geçerek önceden hazırlanmış olması diğer yönteme göre belirgin bir performans artışı sağlamaktadır. Bir diğer artısı da güncel verilerin yer aldığı kaynaklarda sorgulama yaparak performans düşüklüğü yaşanmamasını sağlamasıdır.

Özetle, veri ambarı stratejik kararları verme konusunda bir kurumun ihtiyacı olan bilgiyi depolayan ve karar destek veri modelinin fiziksel bir sunumu gibi çalışan, anlamsal olarak tutarlı bir veri deposudur. Veri ambarı aynı zamanda, yapısal veya planlanmamış sorgular, analitik raporlar ve karar vermeyi desteklemek için farklı türde kaynaklardan veriyi bütünleştirerek oluşturulan bir mimari olarak da görülür (Han and Kamber, 2001).

2.2 Veri Ambarı ile OLTP Sistemleri Arasındaki Farklar

Çevrimiçi hareket işleme adı verilen OLTP sistemleri, organizasyonlarda veri giriş ve çıkışının hızlı olması için tasarlanan ilişkisel veritabanı sistemleridir.

Bir kurumun operasyonel işlemlerini gerçekleştirdiği OLTP sistemler (veritabanları) bilgi toplama üzerine (kayıt ekleme, çıkarma, silme ki bunlar hareket/transaction olarak bilinir) uzmanlaşmıştır. OLTP işlemler Şekil 2.1’de görüldüğü gibi aynı anda birden çok kullanıcı tarafından hareket işlemleri yapılabilmektedir. OLTP veri tabanları; sürekli yeni veriler eklenen, verilerin modifiyesinin yapıldığı durumlar için oluşturulurlar. Bu tür veri tabanlarında bulunan veriler dinamik bir yapıya sahiptirler (sürekli değişmektedirler). Bu sistemler, bir organizasyona ait alım satım, envanter, üretim, bankacılık, ücretlendirme, kayıt ve hesaplama gibi günlük işlemlerin çoğunu karşılamaktadır.



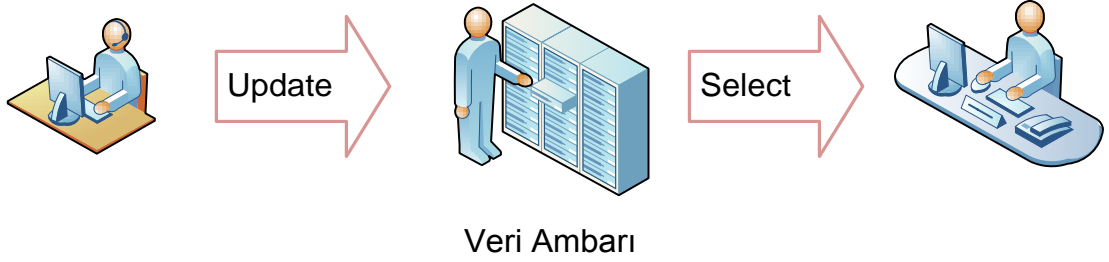
Şekil 2.1 OLTP Sistemlerdeki İşlemler

OLTP veri tabanları verinin hızlı ve etkin bir şekilde girişi, çıkışı ve güncellenmesi için tasarlanmış yapılar olmaları nedeniyle bu veriler üzerinde veri analizi yapmak sistem üzerinde işlem yapan kullanıcıların işlemlerini yavaşlatabilir veya zaman aşımı (time out) hatası nedeniyle işlemleri iptal olabilir. Bu tür olumsuzlukların önüne geçebilmek için yeni bir veri tabanı oluşturmak ve buraya sadece analizde kullanılacak verilerin taşınmasını sağlamak gerekliliği ortaya çıkmıştır. Online işlemlerin gerçekleştiği veritabanlarında veri tutarsızlıkları, kirli veriler, gürültülü veriler gibi analizi engelleyecek ve yanlış sonuçlar çıkarabilecek durumları da engelleyebilmek için etkin bir yöntem olan bu anlayış veri ambarı denilen yeni veri tabanlarının doğmasına neden olmuştur.

İlişkisel veri tabanlarının kullanımı ve sonrasında ortaya çıkan veri ambarlarının büyüklüğü ile beraber, verilere daha hızlı şekilde erişme ve çok boyutlu analiz ihtiyaçları

doğmuştur. Çevrimiçi Analitik İşleme adı verilen OLAP (Online Analytical Processing) veritabanları karar destek sorgularını kolaylaştırır.

OLAP sistemleri Şekil 2.2’de görüldüğü gibi OLTP sistemlerin farklı olarak işlemleri işlemek yerine veri ambarında barındırdığı verileri belirli periyotlarda güncelleyerek sorgulama ve raporlama için kullanılır.



Şekil 2.2 Veri Ambarı Sistemindeki İşlemler

OLTP sistemi müşteri merkezli tasarlanmaktadır. Sisteme erişen kullanıcı sayısı çok fazladır. Bilgi işlem ile ilgilenenler, satıcılar ve müşteriler tarafından veri giriş çıkışı ve sorgulama için kullanılmaktadırlar. OLAP sistemi ise pazar temelli olarak tasarlanmaktadır. Karar alıcı kademede bulunan belirli sayıdaki kullanıcılara yöneliktir. Analizcileri uzmanları ve yöneticileri içine alan karar destek mekanizmasında çalışan kullanıcılar tarafından veri analizi için kullanılır.

OLTP sistemleri, tipik olarak karar vermede kolaylıkla kullanılamayacak seviyede detaylı güncel veriyi içermektedirler. Bir OLAP sistemi büyük miktarlarda tarihi veriyi yönetmektedir. Özetleme ve birleştirmede kolaylık sağlamaktadır. Öğe boyutunun farklı seviyelerinde bilgiyi saklamakta ve yönetmektedir.

OLTP sistemi genelde varlık bağıntı veri modelini ve uygulama merkezli veri tabanı uygulamaktadır. Tasarım sırasında tablolar yüksek seviyede normalize edilmiştir. Böylelikle son kullanıcıların işlemlerinin hızlı yapılması sağlanmıştır. OLAP sistemi tipik olarak yıldız ya da kar tanesi veri modelini amaca yönelik bir veri tabanı tasarımını uygulamaktadır. Tasarım sırasında tablolar denormalize edilir.

Bir OLTP sistemi temel olarak, organizasyonların tarihsel verisinden ziyade ilgili bölümün dâhilindeki güncel veri üzerinde odaklanmaktadır. Bunun tersi olarak bir OLAP sistemi veri tabanı şemasını, organizasyon içindeki işlemlerin evrimsel gelişimine göre, sıklıkla çoklu parçalara bölmektedir. OLAP sistemleri ayrıca farklı organizasyonlardan gelen verilerle de ilgilenmekte ve birçok veri kaynağından gelen veriyi birleştirmektedir.

OLTP sisteminde performans göz önünde bulundurularak işlemlerin hızlı olmasında kullanılan normalizasyon sistemin de etkisiyle karmaşık sorguları yaratmak ve işleme koymak zordur. Bu karmaşık sorguların sonuçlarının dönmesi saatleri bulabilir. OLAP sisteminin yapısından dolayı karmaşık sorgulara gerek kalmadan ihtiyaç duyulan verilere kolayca erişilebilir. Bu bağlamda sorgular kısa sürede cevaplanacaktır.

OLTP sistemler ihtiyaca göre tasarlandıkları için zaman içerisinde gelişmeleri kaçınılmazdır. Bu nedenle birbirinden bağımsız tasarlanan uygulamalar tekrarlı veri oluşumunu doğurur. Sisteme girilmiş her türlü veri mevcuttur. Kullanıcı hatalarından kaynaklanan eksik veya yanlış bilgiler bulunabilir. Hatalı ve tekrarlı veriler nedeniyle kayıtlar analiz için gereksiz veri içerecektir. OLAP sistemler uygulama katmanında geliştirildikleri için gereksiz ve tekrarlı veri girişi engellenir. Sadece ayıklanmış ve veri kullanılması nedeniyle analiz işlemleri için girilmiş her kayıt büyük bir önem taşımaktadır.

OLTP sistemleri işletmelerin anlık verileri içerdiği için yedekleme ve kurtarma planlarının tasarlanması önem arz etmektedir. İşletmelerin bu verileri kaybetmemeleri çok önemlidir. OLAP sistemlerinde veriler analiz işlemlerini için kullanıldığından veri kayıpları kurtarma işlemlerine gerek yoktur. Bu veriler tekrar oluşturulabilir.

2.3 Veri Ambarı Mimarisi

Şekil 2.3’de görülen veri ambarı mimarisinin üç farklı kullanımı vardır. Bunlar Kurumsal Veri Ambarı, Veri Pazarı ve Sanal Veri Ambarından oluşmaktadır.

2.3.1 Kurumsal Veri Ambarı

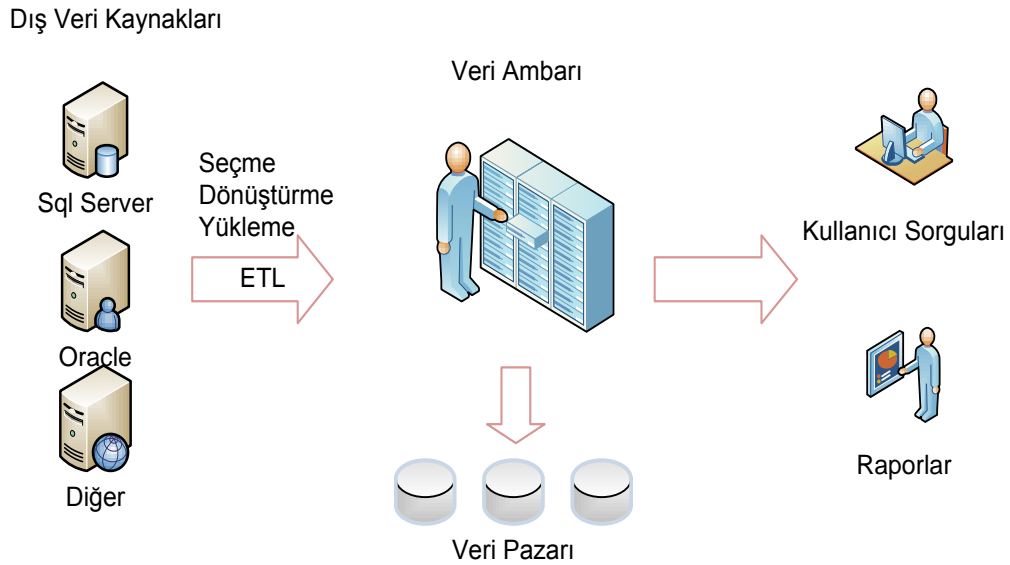
Kapsamlı bir model gerektiren kurumsal veri ambarları organizasyonun tamamına ilişkin özetlenmiş veya detaylı verinin toplandığı ambarlardır. Tüm organizasyon için kullanılacak olan böyle bir sistem modeli geliştirmek, bunu besleyecek çeşitli kaynaklardan çekilecek verilerin ortaya konması uzun bir süre gerektirir ve esnekliği azaltır.

2.3.2 Veri Pazarı (Data Mart)

Veri ambarlarında tutulan veriler işletmelerin operasyonlarından elde edilmiş olan ancak farklı bir yapıda tutulan enformasyonel verilerdir. Bu verilerden bazılarının her bölümün kendi kullanım amacına hizmet edecek şekilde ayrılmasıyla veri pazarları oluşmuştur. Veri pazarı belirli bir konuya odaklı ya da bölüm düzeyindeki bir veri deposudur. Kurumsal veri ambarına göre tabandan tepeye yaklaşım daha kolay uygulanmaktadır. Bazı şirketler genel ve bütünsel bir veri ambarı kullanmak yerine bölüm düzeyindeki veriyi kullanmak isterler. Veri pazarı belirli bir konu veya bölüm bilgilerine odaklanırken, veri ambarı bütün şirketin bilgilerine odaklanır.

2.3.3 Sanal Veri Ambarı

Sanal veri ambarları, güncel olarak kullanılan operasyonel veritabanları üzerinde etkili sorgulama yapılabilmesi için oluşturulan özet veri kümeleridir. Oluşturması en kolay veri ambarı türüdür. Ancak veriler yine operasyonel veritabanı üzerinde tutulacağından çok fazla kapasite gereksinimi ortaya çıkar.



Şekil 2.3 Veri Ambarı Mimarisi

Veri ambarı sistemleri geliştirmede öncelikle en üst seviyede ortak bir veri modeli ortaya konması ve bölümsel veri pazarları tarafından da kullanılması önerilmektedir. Bu sayede veri ambarı ve veri pazarları aynı temeli paylaşırlar ve karşılaşılması muhtemel birçok tümlene probleminin önüne geçilmiş olur.

Yapısal olarak bakılacak olursa veri ambarı üç katmanlı bir mimariye sahiptir.

En alt katmanda ilişkisel veritabanı bulunur. Operasyonel olarak kullanılan veritabanları bu katmanda yer alır. Birden çok dış kaynaktan gelen veriler temizlenip güvenli hale dönüştürülerek bu sunucu üzerinde depolanır. ODBC (Open Database Connectivity), OLEDB (Object Linking and Embedding, Database), JDBC (Java Database Connectivity), Oracle Open Connect gibi standartlar yardımıyla bu kaynaklar üzerinde çalıştırılan sorgular ile veri alımı gerçekleşir.

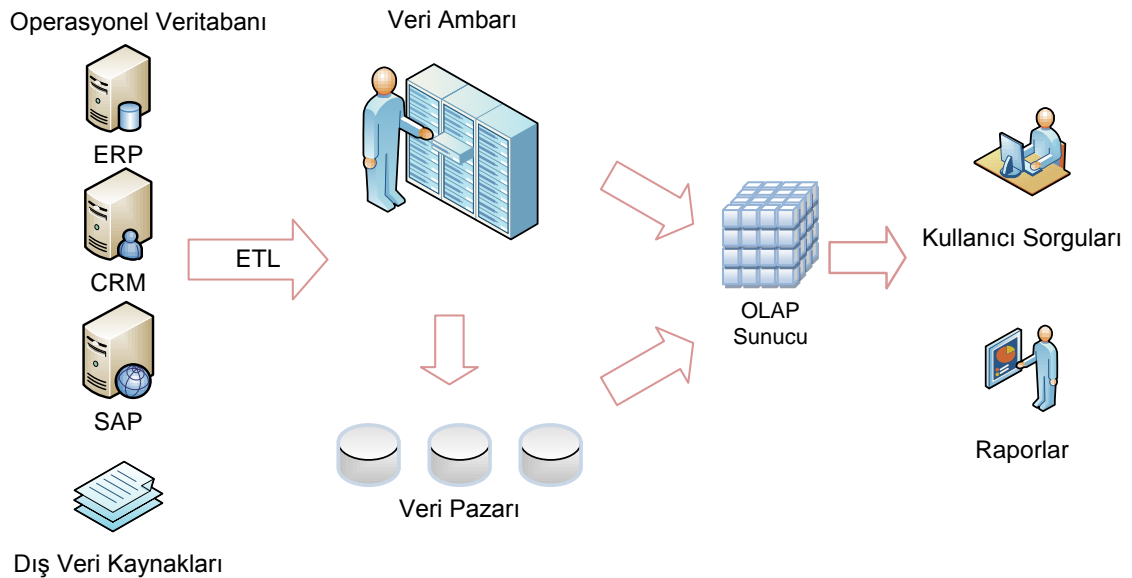
Orta katmanda ilişkisel veritabanında tutulan temizlenmiş veriler üzerinden çok boyutlu verilerin üretildiği katmandır. ROLAP (Relational OLAP, ilişkisel OLAP) veya MOLAP (Multidimensional OLAP, çok boyutlu OLAP) olarak adlandırılan eşleme mekanizmaları yer alır (Seidman, 2001).

En son katmanda karar destek kapsamında son kullanıcılar için sorgu, rapor ve analiz araçları ile bağlantı sağlanır.

2.4 Veri Ambarı Tasarlanması

Veri ambarının operasyonel veri tabanından ayrı olarak tutulan ve karar mekanizmalarına destek için tasarlanan bir veri tabanı olması nedeniyle birçok organizasyon, kurumun tümünü içeren kurumsal veri ambarı oluşumunu ister. Fakat bu kadar karışık bir sistemi inşa etmek ve kullanmak, yapılan işin analizini, teknolojik araçların yeteneklerini bilmeyi gerektirdiğinden karmaşık bir iştir.

Veri ambarı tasarımını yaparken, karar destek ihtiyaçlarını karşılamak için farklı veri kaynaklarındaki verilerin veri ambarına entegre edileceği unutulmamalıdır. Bu nedenle bu işi yapacak kişilerin analiz ihtiyaçları, veri büyüklüğü ve performans kriterlerini göz önünde bulundurup iş gereksinimini ortaya çıkarması, analiz etmesi ve analiz sonucunda optimum bir yapı ortaya çıkarması gerekmektedir. Sistemin tasarımcısı, kullanıcılar ve karar vericiler sisteme farklı açılardan baktıklarından bu bakış açılarının birleştirilmesi sonucu bir sistem ortaya konmalıdır.



Şekil 2.4 Veri Ambarı Bileşenleri

Tasarım sürecinde yukarıdan aşağıya (top down) veya aşağıdan yukarıya (bottom up) yaklaşımı kullanılabilir. Yukarıdan aşağıya tasarım modelinde ayrıntılı bir tasarım ve planlama ile başlar. İş problemlerinin açık ve iyi anlaşıldığı durumlar için kullanılan bu model planlı bir şekilde işin yapılmasını ve sorunların önceden görülmesini sağlar. Bu sistemin kullanılması ile bütünleştirme problemleri minimize edilmiş olur. Yukarıdan aşağıya modelinde tüm organizasyon için tutarlı ve ortak bir veri modeli oluşturmak esnekliği azaltacaktır. Bu yönteme alternatif olarak veri pazarlarının oluşturulmasını sağlayan aşağıdan yukarıya doğru tasarım yöntemi ortaya çıkmıştır. Bu yaklaşımda hızlı uygulanabilirlik önemli bir avantaj sağlar. Aşağıdan yukarıya doğru tasarım yöntemiyle organizasyonun tümü yerine organizasyon içerisindeki alt bölümler için küçük veri ambarlarının her biri veri pazarlarını oluşturur. Bu yaklaşım deneyler ve prototiplerle başlar. İş modellemenin ve teknoloji gelişiminin erken aşamalarında kullanışlıdır. Aşağıdan yukarıya doğru yaklaşım esneklik, düşük maliyet ve yatırımların kısa zamanda geri dönüşünü sağlar. Her iki yöntemin de kullanıldığı birleşik yaklaşımda, organizasyon aşağıdan yukarıya doğru yaklaşımın hızlı uygulaması ve fırsatlar sunan kullanımına sahipken, yukarıdan aşağıya doğru yaklaşımın planlı ve stratejik özelliğini de kendi yararına kullanabilir (Kimball, 2002).

Veri ambarının oluşturma sürecinde izlenmesi gereken temel aşamalar şunlardır;

- Kaynak sistemlerin belirlenmesi: Günlük işlemler sonucu ortaya çıkan, işletimsel veri tabanlarında veya diğer dış veri kaynaklarında tutulan verilerden uygun olanlar veri ambarları oluşturulmasında kullanılmak üzere belirlenir.
- Veri ambarı tasarım ve yaratılması: Veri ambarının modelleme sürecinde, veri ambarından yararlanacak sorgulara cevap verebilecek bir yapı oluşturulmasına özellikle önem verilmeli ve dikkat edilmelidir. Tasarım genellikle adım adım ilerleyen bir süreçtir. Oluşturulan model defalarca güncellenebilir. Bu süreçte çok dikkatli olunmalıdır, çünkü veri ambarı çok büyük boyutta veri atılarak hayata geçirilir ve bu andan sonra modeli yeniden oluşturmak veya değiştirmek çok güçleşir.
- Veri oluşturulması: Verinin kaynak sistemlerden ambara taşınması işlemidir. Genellikle veri ambarı projesinin en çok zaman harcanan ve çaba sarf edilen kısmıdır ve ETL (Extract Transform Load) adı verilen yazılım araçları ile yapılır.

Veri oluşturulması veri ambarının sürekli güncel tutulabilmesi için çizelgelenmiş olarak sürekli tekrarlanır.

- Değişen veri yakalama: Veri ambarının düzenli aralıklarla güncellenmesi, operasyonel sistemde son güncellemeden bu zamana değişen verileri yakalamayı güçlüğü nedeniyle karmaşık bir iştir. Değişen verinin yakalanabilmesi için Replikasyon sunucuları, veritabanı tetikleyicileri, saklı prosedürler ve veritabanı log analizleri gibi yöntemler kullanılır.
- Veri temizleme: Genellikle veri oluşturmasına sırasında yapılır. Veri formatının düzenlenmesinden ziyade alanların gerçek içeriklerinin dönüştürülmesi işlemidir. Doğru olmayan veri içeren veri ambarları amaçsız olduğu gibi tehlikelidir de. Veri ambarını arkasındaki tek fikir karar almaya destek olmaktır. Doğru olmayan veriye dayanan üst yönetim kararları işletmeyi zor durumlara sürükleyebilir. Veri temizlenmesi karmaşık bir işlemdir ve verinin ambara atılmadan önce doğrulanması ve gerekiyorsa düzeltilmesini içerir.
- Veri özetlenmesi: Bu işlem genellikle veri oluşturulması sırasında veri dönüştürülmesi ile yapılır. Veri ambarları her bir veriyi tutacak veya belirli detaya göre özet veriyi tutacak ya da her ikisini bir arada tutacak şekilde tasarlanabilir. Özetlenmiş veri ile veri ambarı daha hızlı çalışır fakat ileride lazım olacak bazı bilgiler kaybolabilir. Bu noktada verinin detaylılığına karar verilirken dikkatli olunmalıdır. Çünkü geri dönmek veya veriyi yeniden oluşturmak mümkün olmayabilir. En iyisi veri ambarını olabildiğince detaylı tasarlamaktır.

2.5 Veri Ambarı Uygulama Örnekleri

Aköz, 2007 yılında “Otomotiv Sektöründe Veri Ambarı ve Bir Uygulama” isimli çalışmasında, veri ambarı bilgi teknolojisi Microsoft SQL Server 2000’in işlevselliğinden yararlanarak Doğu Otomotiv Servis Ticaret A.ş. Satış Sonrası Hizmetleri operasyonel verilerine uygulanan bir Veri Ambarı oluşturmuştur. Çalışmasında; Doğu Otomotiv Servis Ticaret A.ş. Satış Sonrası Hizmetlerde mevcut operasyonlarında kullanılan veri akışı ve

yönetimi incelenmiş, veriler toplanıp, bu bilgilerin Veri Ambarına dönüştürülmesi ve üzerinde çeşitli analizler yapılmasına olanak sağlayan raporlar tasarlamıştır.

Güratan, 2005 yılında yaptığı çalışmada bir perakendeci firmasının ihtiyaçları ve satış veri tabanı kullanarak bir veri ambarı tasarlayıp, geliştirmiştir. Firmanın Satış Saha Organizasyonu tarafından toplanan veriler kullanılarak Satış ve İnsan Kaynakları Bölümlerine destek olması amacıyla veri ambarı tasarlanmıştır. Veri ambarının oluşturulmasının ana amacı, saha organizasyonunda görevli kişilerin çalışmalarının satış şefleri tarafından kontrolünün yapılabilmesini sağlamak ve İnsan Kaynakları Bölümünün hazırlayacağı performans değerlendirme sistemi için tutarlı ve doğru verilerin toplanmasını sağlamaktır. Proje gerçekleştirilirken veri transfer ve analiz araçlarını da içinde bulunduran Microsoft SQL Server 2000 kullanılmıştır.

Çetinyokuş ve Gökçen 2007 yılında yaptıkları çalışmada Microsoft SQL Server 2005 kullanılarak Nortwind satış veritabanından satış veri ambarı oluşturmuştur. Oluşturulan veri küpünde boyutlar; zaman, ürün ve lokasyon, hesaplanmış değerler ise satış toplamları bilgisidir. Küpteki veriler küp yapısına uygun olan çok boyutlu dizilere aktarılmıştır. Daha sonra bu mimarideki veriler kullanılarak ürün ve lokasyonlar bazında regresyon modelleri oluşturulmuştur.

Gray tarafından 1997'de küp işlemleri Yapısal Sorgulama Dili (Structured Query Language)'nin gruplama (GroupBy) işlemcisinin geliştirilmesiyle ortaya çıkarılmıştır. Küp belirlenmiş boyutların tüm alt setlerindeki veri setlerinin gruplama sonuçlarını hesaplayabilmektedir. Böylece 2d kadar grup oluşmaktadır. "d", 2d ifadesinde küpün boyut sayısını göstermektedir. Her bir grup küboid (cuboid) ve tüm sonuçlar ise küp olarak kullanılmaktadır.

Aggarwal ve van der Valk Microsoft SQL Server 2005 ile veri ambarı ve OLAP uygulamaları üzerinde testler yapmışlardır. Bu test çalışması için Avrupa'da önde gelen bir marketler zincirinin verileri kullanılmıştır. Microsoft SQL Server 2005 Analysis Servis yüklü 32 çekirdekli Unisys ES7000 sunucuda 1.3 milyar veri yaklaşık olarak 21 dakikada işlem görmüştür. Aynı işlem 64 çekirdekli sunucu için yapıldığında 2.1 milyar veri 19 dakikada işlem görmüştür.

Kuğu, Oracle Olap teknolojisi kullanılarak bir filodaki faal uçak sayısının uçuş saatlerine etkilerinin gözlemlenmek üzere kullanılacak bir boyutsal model tasarlanmıştır.

Güleç, 2007 yılında yaptığı çalışmada işletmelerin mevcut verilerinden anlamlı bilgilerin üretilmesine yönelik modelleme aracı tasarlanmıştır. Hazırlanan bu araç ile veri madenciliğinde kullanılan farklı tekniklerin ortak bir veri ambarını ortamında toplanması sağlanmıştır. Tekniklerin bir araya toparlanması yaklaşımı ile işletmelerin kurumsal verilerden ilgilenen bilgileri daha hızlı ve daha az uzmanlık bilgisiyle elde edebilmeleri amaçlanmıştır.

Döşlü, 2008 yılında market sepet analizinde kullanmak üzere çok büyük miktardaki ham veriden değerli bilginin çıkarılması işlemi için veri ambarı tasarlamıştır. Veri ambarı tasarımı için Microsoft SQL Server 2005 kullanılmıştır. Bu çalışmada ele alınan market sepet analizi veri ambarı yardımıyla veri madenciliğinde kullanılan algoritmalar kullanılmıştır.

Birant, 2006 yılında yaptığı çalışmada bilgisayar bilimleri ve deniz bilimlerini birleştirmiştir. Deniz verilerinin modellenmesi, toplanması, arşivlenmesi için veri ambarı tekniklerinin kullanılmış ve analizlerin yapılması deniz verilerinin görselleştirilmesi için yeni bilgisayar tabanlı yaklaşımlar, teknikler ve teknolojiler geliştirmiştir.

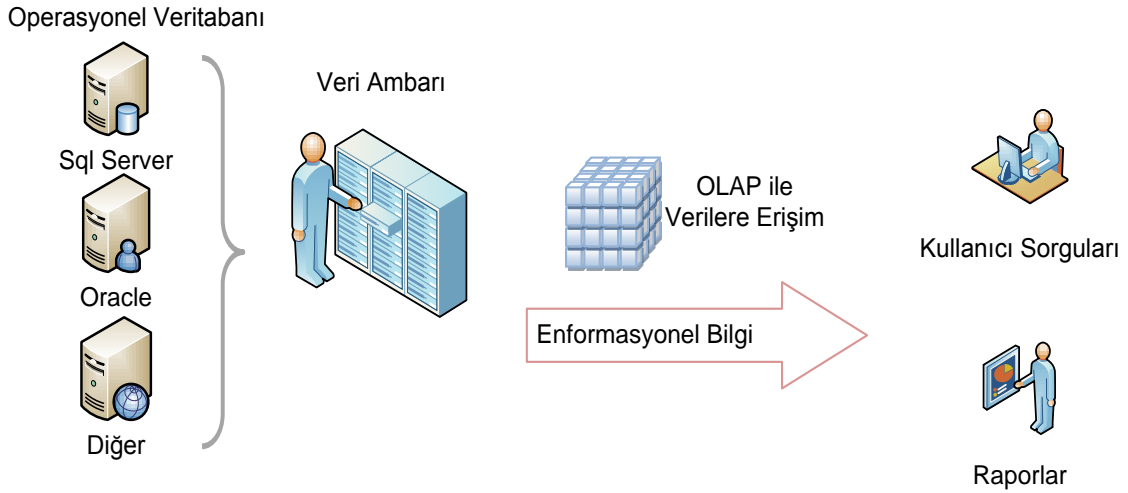
3. OLAP SİSTEMİ

3.1 Giriş

Veri ambarında veri oluşturulduktan sonra bu verinin elle veya gözle analizi yapılabilir. Bunun için çok boyutlu veri analizlerini destekleyen sorgulama temelli bir metot olan OLAP teknolojisi kullanılır. OLAP, uzman kişinin sisteme olan bakış açısını sistemi kullanan kişilerin anlayacağı biçimde ifade ederek, ham veriden dönüştürülmüş bilgi üzerinde mümkün olan incelemelerin büyük çoğunluğuna hızlı, tutarlı, interaktif biçimde erişip analizcilere, yöneticilere ve uygulayıcılara veriden çeşitli kavrayışlar sağlayan bir yazılım teknolojisidir. Veriye her boyutu bir alana karşılık gelen çok boyutlu bir küp olarak bakmayı ve incelemeyi sağlar. Böylece boyut bazında gruplama, boyutlar arasındaki bağıntıları inceleme ve sonuçları grafik veya rapor olarak sunma olanağı sağlar.

Bir veri ambarının olması, OLAP'a ihtiyaç olmadığı anlamına gelmez. Veri ambarları ve OLAP birbirlerinin tamamıdır. Veri ambarı verileri barındırmaya yarar. OLAP ise, bu yığın halinde duran verileri anlamlı hale getirip analizler yapmaya yarar.

OLAP sisteminin kullanıcılara kattığı en önemli kazanç tartışmasız zaman kazancıdır. Buradaki zaman kazancı ilişkisel veri tabanlarındaki raporlamaya kıyaslanamayacak kadar büyüktür.



Şekil 3.1 Veritabanı ve Veri Ambarı ilişkisi ve Bilgi Mimarisi

OLAP sistemleri verilere çok boyutlu bakabilme özelliğini içerisinde barındırmaktadır. Analizler sırasında kullanmış olduğumuz, her türlü kırılıma boyut adını verebiliriz. Yöneticiler ve analistler, çalışmalarını sırasında, tüm bu tanımlanan verileri yatay veya düşey eksenlerde çakıştırarak görmek isterler. İlişkisel veri tabanları, bu şekilde raporlara izin vermezler, fakat raporlama araçlarının yetenekleri ile belirli bir noktaya kadar çok boyutlu olmayan OLAP benzeri raporlar alınabilir. Fakat daha karmaşık analizler hazırlanması gerektiğinde, bir OLAP yapısı kurmadan bu raporları almak imkânsız hale gelebilir.

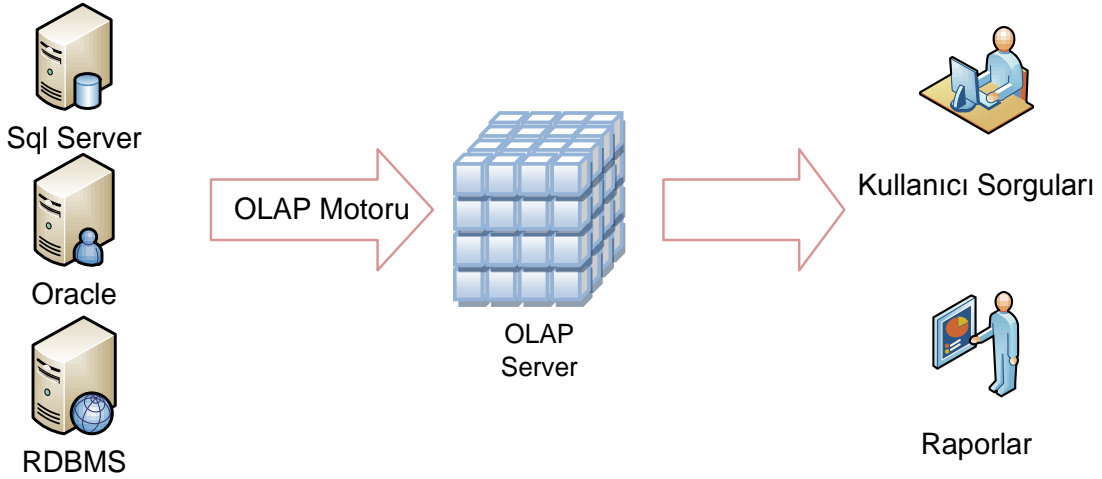
İlişkisel veri tabanları üzerinde karmaşık SQL (Structured Query Language- Yapılandırılmış Sorgu Dili) kodları yazmak, ya da raporlama aracının sahip olduğu programlama dili üzerinde uğraşmak gerekebilir. Bu da, analizi yapan kişilerin, en az bir veri tabanı yöneticisi ya da programcısı kadar bilgiye sahip olması sonucunu doğurur. Bu nedenle, iş zekâsı programlarının pratik olmasının yanında, fazla teknik bilgi kullanmadan raporların alınabilir olması, farklı kaynakları bir arada kullanabilecek, pekiştirilmiş edebilecek yapıda olmaları gerekir. Boyutların başka bir özelliği de hiyerarşiler tanımlanabilmesidir. Hiyerarşiler sayesinde, hem toplamlara ulaşmak kolaylaşmakta, hem de farklı gruplar için, farklı senaryolar hazırlayabilme şansı doğmaktadır.

Bir OLAP sisteminin gerçek performansı, karmaşık hesaplamaları yapma gücü ile ölçülebilir. OLAP sistemleri, sadece toplama işleminden başka işlemler de yapabilecek

güçte olmalıdırlar. Gerçek hayat, her zaman daha karışıktır. Analiz yapanlar için, asıl rakamlardan çok, yüzdesel dağılımlar çok daha önemlidir. Birkaç yıllık satış içerisinde, binlerce ürün türü için günlük bazda satışları yüzdesel olarak analiz edip, sıraya dizebilmek sql cümlecikleri ile saatler sürecektir bir raporun çalışmasını gerektirebilir. Oysa uygun bir OLAP sistem ile bir günlük satışlar ve birkaç yıllık satış rakamı arasında bir fark yoktur. Satış tahminlerinde, genellikle “yüzde artış” gibi eğilim analizleri kullanılır. Finansal analizlerde, envanter hesaplarında ve portföy performans hesaplarında, zamana göre ürünlerin toplanma sırası, sonucu tamamen değiştirebilir. (yukarıdan aşağıya, ya da aşağıdan yukarıya, LIFO-FIFO) Kullanılacak OLAP yapısında, bu şekilde hesaplamalara da izin verir bir yapısının olması gerekir.

Zaman boyutu, neredeyse her analizin temel bileşenidir. Zaman, diğer boyutlardan farklı olarak, kendine has bir sıralama içerisinde gider. Alfabetik (Ocak her zaman Şubat’tan önce gelmelidir) veya nümerik sıralamalardan (31.12.2008, 01.01.2009’dan önce gelmelidir) her zaman farklıdır. OLAP sistemleri, zamanın bu şekilde sıralanmasını sağlarlar.

Veri Ambarı



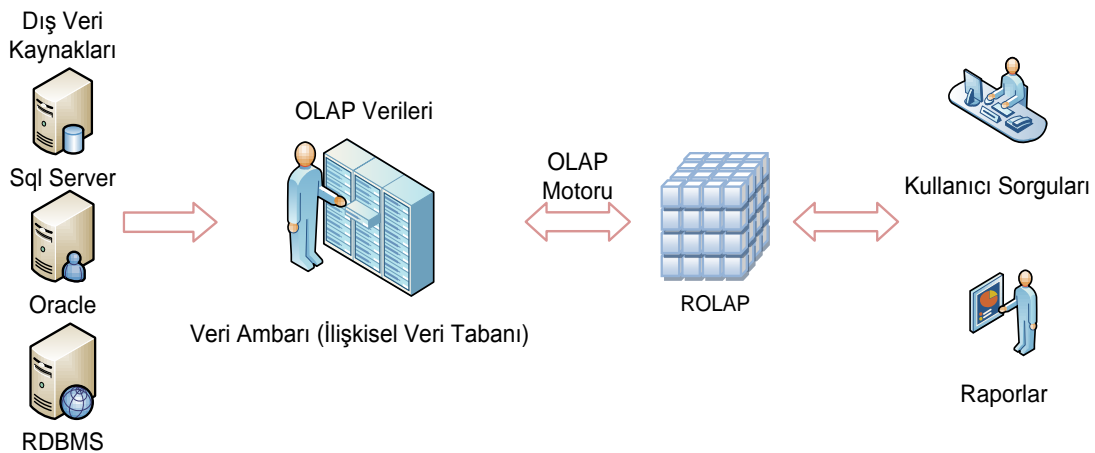
Şekil 3.2 OLAP Mimarisi

3.2 OLAP Sunucu Türleri

OLAP kapsamında üç sunucu türü mevcuttur (Seidman C, 2001). Bunlar sırası ile aşağıda tanımlanmıştır.

3.2.1 ROLAP

Şekil 3.3’de ROLAP modelinde veri saklama işlemleri için orijinal ilişkisel veritabanı kullanılır. Bilgiler veritabanındaki tablolarda tutulur. Çok boyutlu operasyonlarda kullanılacak verilerin ilişkisel tablolarda tutulması özel mekanizmalara gerek duymadan standart sorgulama işlemleri kullanılarak raporların oluşturulmasına olanak sağlar. İş yükünün büyük kısmının ilişkisel veritabanında olması nedeniyle OLAP sistemi üzerinden oluşturulacak sorguların çalıştırılması oldukça zaman alacaktır.

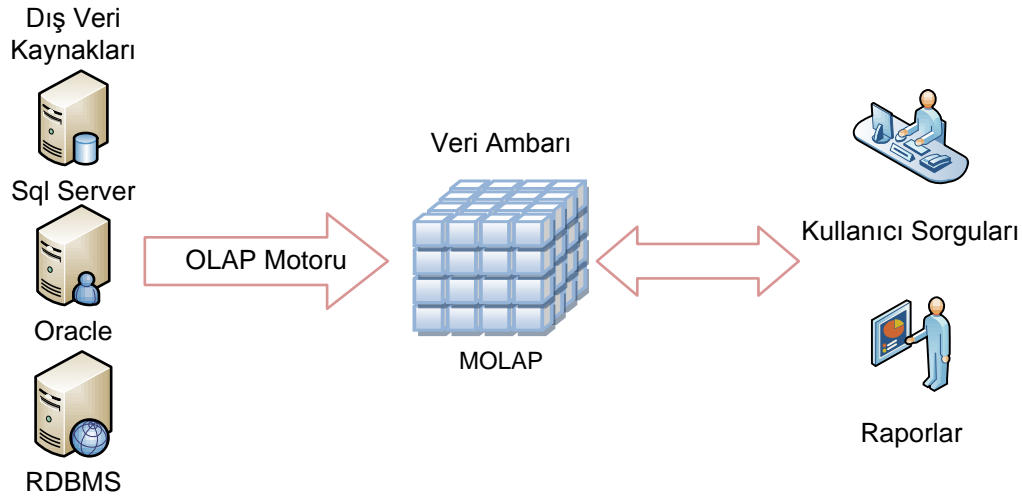


Şekil 3.3 ROLAP Mimarisi

3.2.2 MOLAP

Şekil 3.4’de MOLAP modelinde çok boyutlu veri küplerin veritabanı haricinde dış bir kaynak olan OLAP sunucusunda tutulur. Çok boyutlu verilerin dizi gibi özel veri yapıları üzerinde tutulmasını sağlar ve OLAP işlemleri bu özel veri yapılarına erişilerek

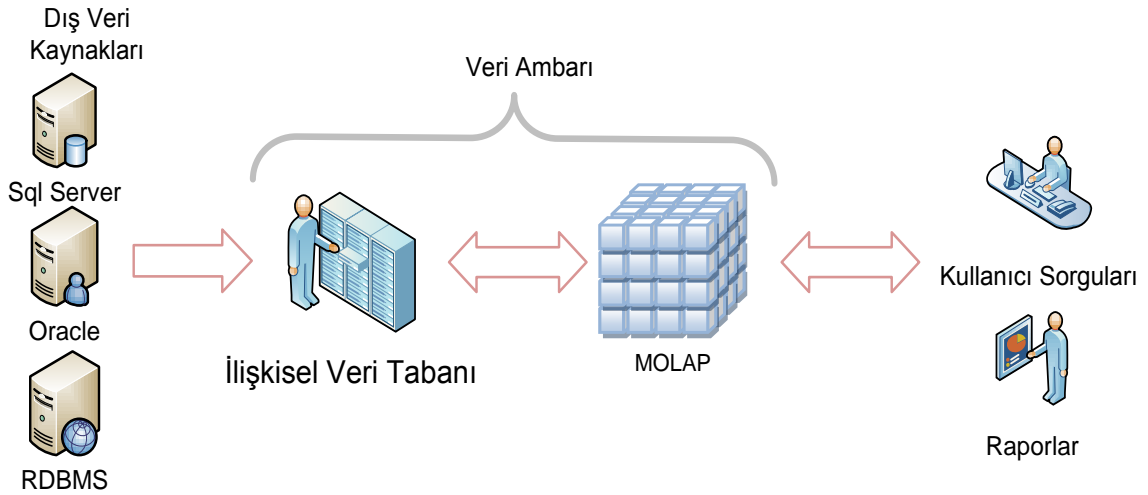
gerçekleştirilir. Tanımlanmış hesaplamalar için veriler özetlenir ve işletilir. Bu yöntem en iyi sorgulama performansını sağlar. MOLAP yönteminin kötü tarafı, küpün veri ambarı ile bağlantısının koparılmış olmasıdır. Veri ambarı bir ilişkisel veritabanı kapsamında yer alırken, küpler özel veri yapıları olarak sistemde saklanır. Bu durum küpün güncellenmesi işlemini oldukça karmaşıklaştırır.



Şekil 3.4 MOLAP Mimarisi

3.2.3 HOLAP

Şekil 3.5'te HOLAP (Hybrid OLAP, melez OLAP) model ROLAP ve MOLAP modellerini kapsar. HOLAP modeli MOLAP ve ROLAP yaklaşımlarının avantajlarını bir arada bulundurur. Bu veri modelinde veriler yine ilişkisel veritabanı üzerinde tutulur. Yüksek seviyeli özetlenmiş veriler MOLAP küpleri ile saklanırken, düşük seviyeli özet verileri ve tablolardaki kayıtlara ilişkisel veritabanından erişim sağlanır. HOLAP kapsamında veritabanında tutulacak tabloların karmaşıklığı olabildiğince azdır. Dolayısıyla, kurulacak indeksleme mekanizmaları ile veriye performanslı bir şekilde ulaşılır.



Şekil 3.5 HOLAP Mimarisi

3.3 OLAP Mimarisi

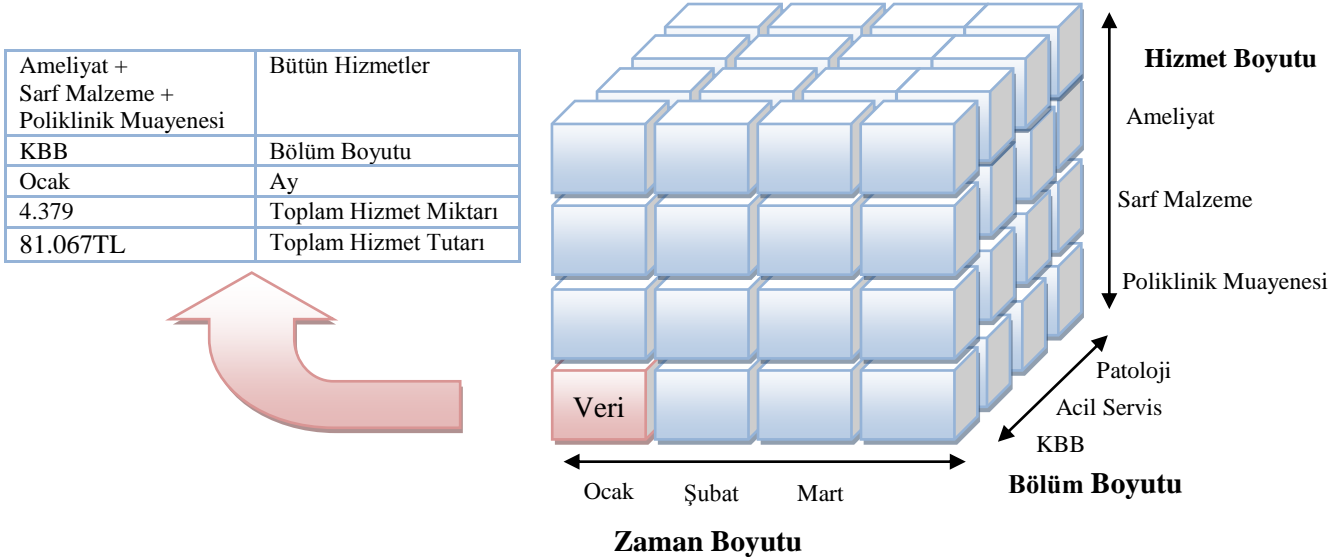
OLTP veritabanlarından çok daha büyük disk alanına sahip veri ambarlarına aktarılan veri buradan da işlenerek OLAP veritabanlarında yer alan küplere aktarılır. OLTP veritabanlarında birçok tablodan ve kayıttan birleştirilerek elde edilen veriler veri ambarında saklanır. Daha sonra veri ambarından çok boyutlu sorgular için optimize edilmiş küp veri yapıları içine (OLAP) aktarılır.

OLAP veritabanı özellikleri;

- Çok kullanıcı desteği
- Performans
- Esnek raporlama
- Çapraz rapor olanağı
- Sorgulamalarda aynı performansı gösterebilme
- Boyutlarda sınır olmamasıdır.

OLAP araçları, verinin çok boyutlu kavramsal görünümünü desteklemektedir. Çok boyutlu modelde, veriler satır ve sütunlar yerine olay (fact) ve boyutlarda (dimension) saklanmaktadır. Verinin çok boyutta modellenmesi ve incelenmesi Veri Küpleri ile sağlanır. Küpler OLAP veritabanlarında kullanılan mantıksal tablolardır. Boyutlarla

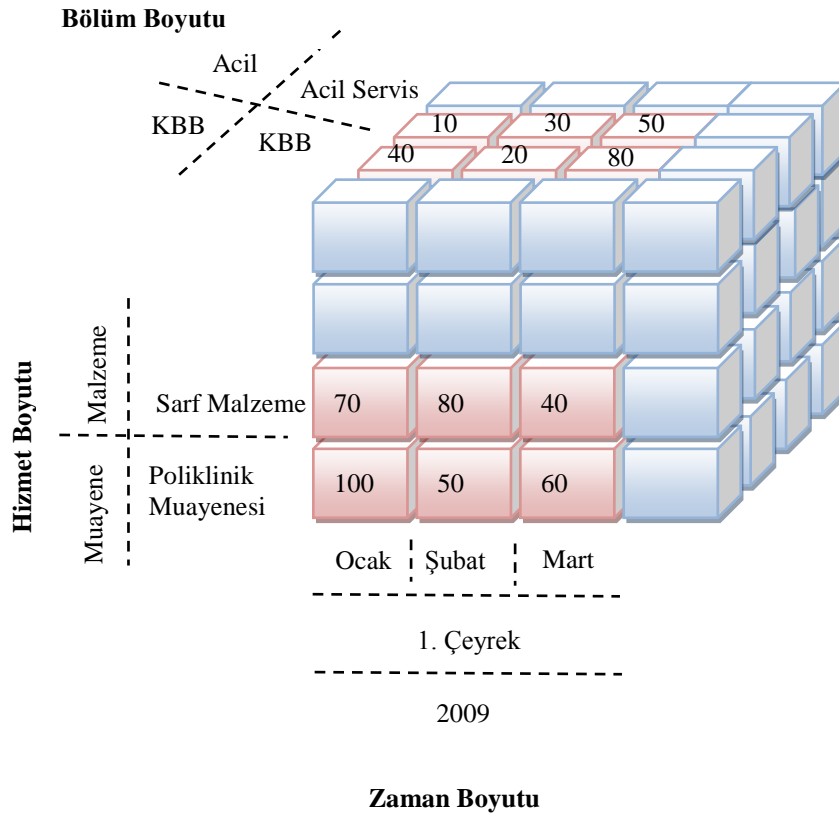
ölçüm değerlerinin bir arada tutulmasını sağlarlar. Küp, içinde birçok boyutu dâhil ederek n-boyutlu ızgaralar oluşturur. Şekil 3.6’da görülen 3 boyutlu bir küp ve şekil 3.7’de üç boyutlu küboid hiyerarşik yapısı verilmiştir. Küpün her bir hücresi sadece bir değere karşılık gelir. Boyutların kesişmesi sonucunda o değere ulaşılır.



Şekil 3.6 Küp Yapısı

Veri küplerini oluşturan temel öğeler aşağıdadır;

- Olaylar
- Boyutlar
- Ölçüm Değerleri



Şekil 3.7 Üç Boyutlu Küboid Hiyerarşik Genel Kavramı

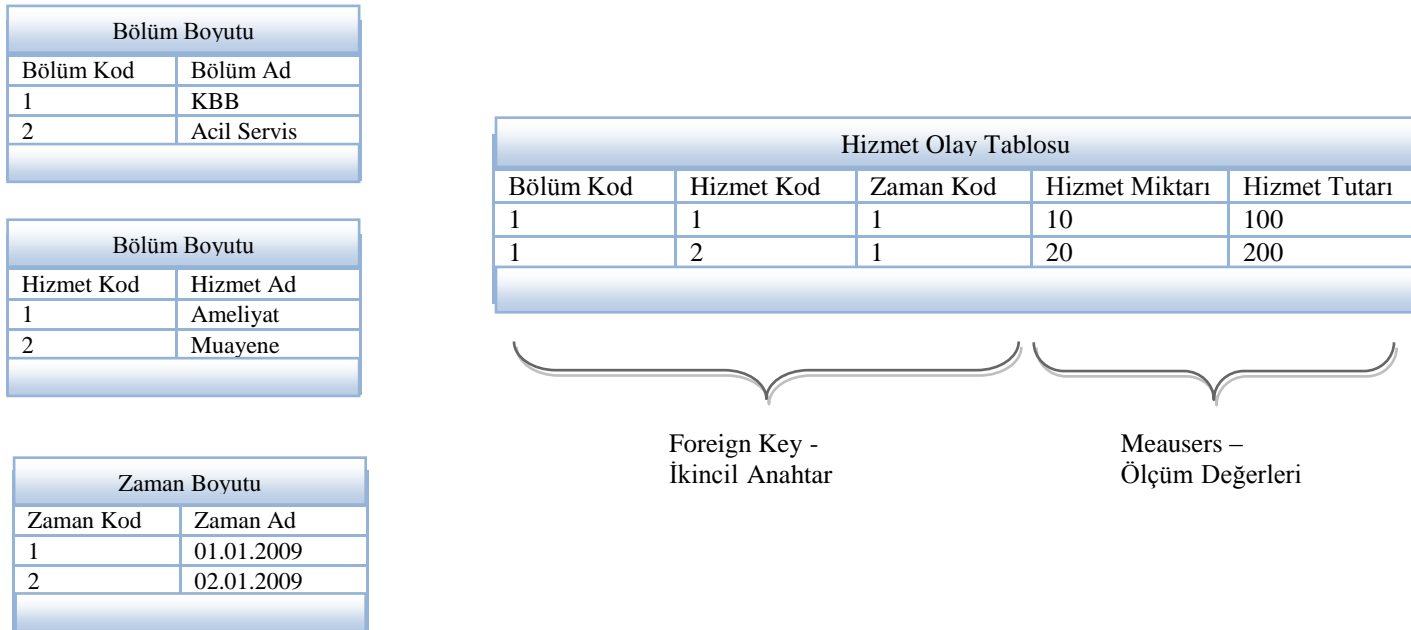
3.3.2 Olaylar

Bir küp oluşturulurken bir ana veri tablosu kullanılır. Bu veri tablosu küpün boyutları aracılığı ile özetlenecek olan ölçü birimlerinin bulunduğu olay (fact) tablosudur. Şekil 3.8’de görüldüğü gibi olay tablosu boyut tabloları ile ilişkilendirilir. Olay tablosu iş içerisinde gerçekleşen spesifik olayları yani iş metriklerini içerir.

3.3.3 Boyutlar

Boyut, genel olarak bir çizelge üzerindeki bir eksen ifade eder. Boyut tablo, olay tabloya tanımlayıcı nitelikler ile ilişkili tablodur. Boyut veri üzerinde tekil bir bakış

açısını simgeler. Tekil ölçüm değeri ya da küpün bir yüzüne karşılık gelir. Boyutlar sayısal olmayan fakat bilgi verici varlıklar olarak tanımlanabilir. Boyut tabloları bilgi tablosunun içeriğini oluşturup sayısal veriyi sağlayan bir iş kaydını gösterir. Boyut tablolarının oluşturulmasının amacı kullanıcının ihtiyacı olan kullanılabilir, tanımlayıcı analitik bilgileri kullanıcıya anlaşılır bir dille sunmaktır.



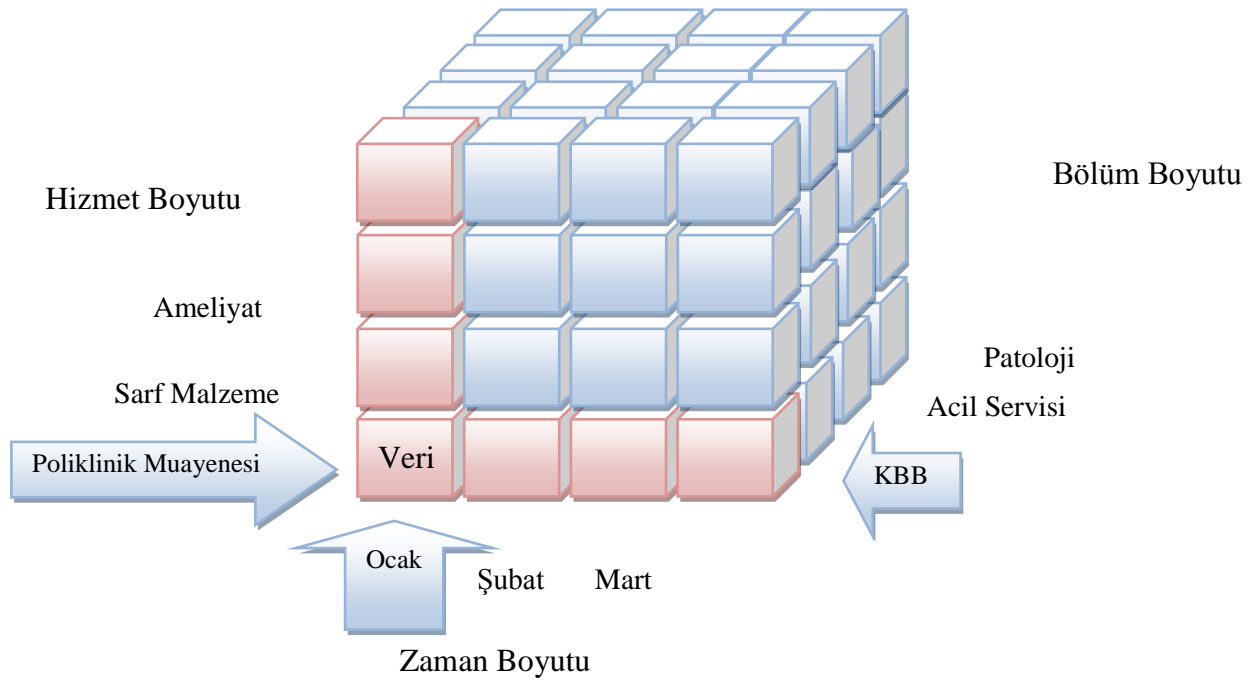
Şekil 3.8 Olay Tablo Yapısı

3.3.4 Ölçüm Değerleri

Ölçüm değerleri bilgi tablosu içerisindeki nicel, sayısal sütunlardır. Analiz edilen sayısal değerleri sunmak için kullanılır. Satılan ürün miktarı ya da müşteri sayısı örnek olarak verilebilir. Ölçüm değerleri genelde değer eklenebilen, sayısal ifadelerdir, çünkü bu sahalarda hesaplamaların yapıldığı merkez sahalardır. Şekil 3.9'da küp içerisinde veriye erişim gösterilmiştir.

Olay tablosu, genellikle büyük yoğunlukta veri taşır. Olay ve boyut tablolarından oluşmuş bir küpe ait verilerin hepsine eş zamanlı olarak erişmek ve bu verileri doğru olarak algılayabilmek mümkün değildir. Bu nedenle, küp üzerinde ikişer boyut seçilerek

verinin anlaşılabilirliği ve kullanıcıya erişim hızı artırılır. Veriye erişimlerde iyi bir performansı yakalayabilmek için tek bir olay tabloya bağlanan boyutların sayısını yönetilebilir bir seviyede tutmak gerekir. Boyutların sayısı arttıkça, olay tablo üzerindeki satırların sayısı da artacaktır.



Şekil 3.9 Küp İçerisinde Veriye Erişim

3.4 OLAP Küplerinde Genel İşlemler

Şekil 3.12’de verilen OLAP veritabanında oluşturulan küplere ait genel özellikler şunlardır:

Roll up, boyutlar üzerinde gruplama seviyesini artırarak daha özet veriye ulaşmayı sağlar. Verinin özetlenmiş haline ulaşılır. Şekil 3.10’da verilen “Hizmet Miktarları” işlem tablosuna uygulanan roll up işlemi sonrasındaki veriler gösterilmektedir

Drill down, daha yüksek ayrıntı düzeylerine inilir. Roll-up işleminin tersidir, gruplama seviyesi azaltılarak çok daha detaylı bilgilere erişilir. Şekil 3.11’de özet olarak verilen “Hizmet Miktarları” tablosu drill down işlemi ile ayrıntılı hale gelmiştir.

Slice ve Dice, seçme ve izdüşümü almaz, verinin boyutu azaltılıp, belirli bir boyutun alt kümesi alınarak izdüşüme erişilir. Slicing (dilimleme) ve dicing (ızgaralama), OLAP küp içerisindeki verinin iş analizlerini belirlemek için, yaygın olarak kullanılan terimdir. Slicing, bir veya birden çok boyut elemanlarını ayırarak tutar ve bu elemanları diğer boyutlara göre değerlendirir.

Hizmet Miktarları							
2009							
		Ocak	Şubat	Mart	Nisan	Mayıs	Haziran
Malzeme	Malzeme	20	30	40	50	60	40
	İlaç	25	35	20	45	65	35



Roll Up

Hizmet Miktarları			
2009			
		Ocak	Şubat
Malzeme	Malzeme	90	150
	İlaç	80	145

Şekil 3.10 Roll Up İşlemi

Pivot, çok boyutlu görüntünün yeniden yönlendirilmesi işlemidir. Çok boyutlu verinin (x,y) koordinatlarıyla ifade edilecek alt parçalara bölünmesini sağlar. Boyutlardan sadece ikisi kullanılır ve boyutların kesişim noktası ölçüm değerini verir. Çok boyutlu verinin iki boyutlu olarak incelenmesini sağlar. Pivot Table kullanıcılara OLAP küplere göz atmak için hassas bir arabirim sağlar. Kullanıcılar veriyi daha çok detaylandırabilirler veya daha özet hale getirebilirler. Ayrıca bu arayüz kullanıcılara slice (dilim) ve dice (ızgara) olanağı sağlar.

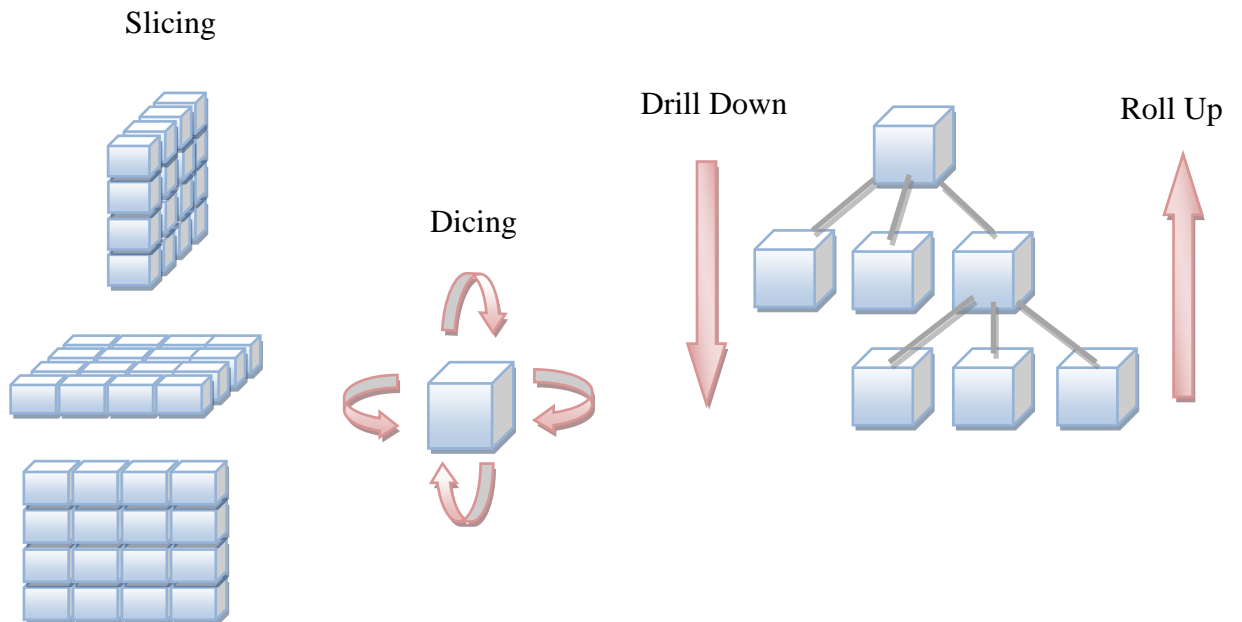
Hizmet Miktarları					
		Malzeme		Muayene	
		Sarf Malzeme	Sarf İlaç	Poliklinik Muayenesi	Acil Muayenesi
Hepsi	DHL	150	170	250	170
	ACİL	250	205	260	190



Drill Down

Hizmet Miktarları					
		Malzeme		Muayene	
		Sarf Malzeme	Sarf İlaç	Poliklinik Muayenesi	Acil Muayenesi
Hepsi	Dâhiliye	150	170	250	170
	Endokrinoloji	50	100	100	90
	Çocuk Acil	250	205	260	190

Şekil 3.11 Drill Down İşlemi



Şekil 3.12 OLAP Küpleri Genel İşlemleri

3.5 OLAP Tasarımı

OLTP sistemlerinde varlık bağıntı diyagram (entity relation diagram) ile veritabanı tasarımı yapılmaktadır. Fakat çok yoğun sorgulama ve veri yükleme içeren karar destek sistemleri için varlık bağıntı diyagramları çok uygun değildir.

Veritabanı tasarımında, OLAP sorgu motorları ve ön uç araçları etkileyen en popüler kavramsal model çok boyutlu görüntülerdir. Çok boyutlu modelde istenilen sayıda ölçüm değerini içeren konu başlıkları vardır.

Çok boyutlu modelde, varlık bağıntı diyagramında yer alan varlıklar boyutlarla eşleştirilir. Birden çok boyuta sahip olan ve ölçülebilir nicel değerleri barındıran varlık, olay tablo olarak nitelendirilir.

OLTP sistemlerinin tersi bir şekilde Normalizasyon kurallarının Veri Ambarlarında uygulanmamasının sebebi veriye farklı bakış açılarından bakarak, daha önce keşfedilememiş olan bilgiye ulaşabilmek içindir.

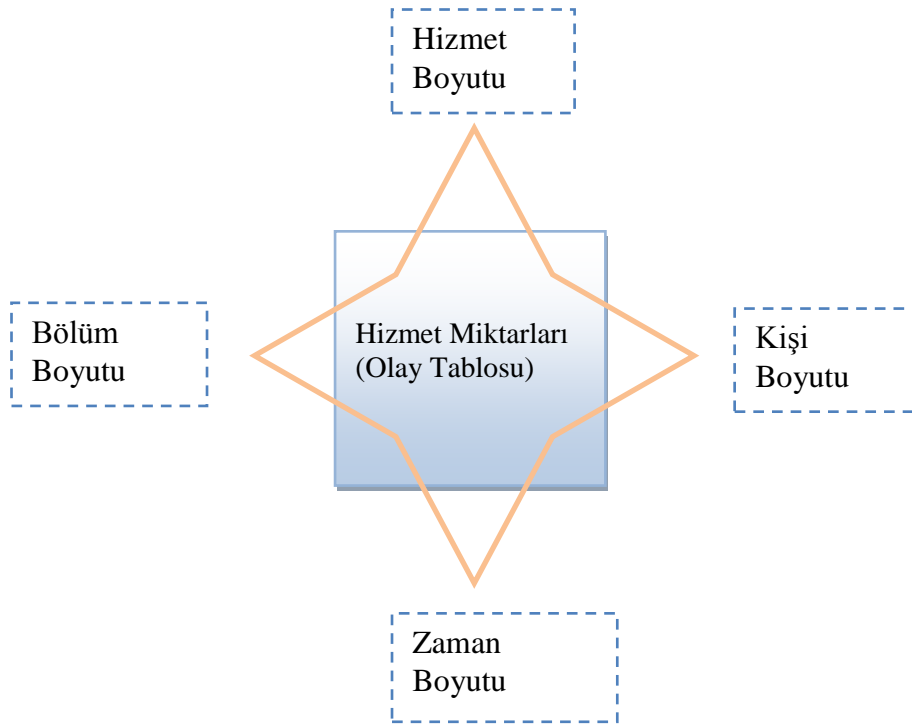
Veri ambarı tasarımında veritabanı özellikleri;

- Yıldız Şema Modeli,
- Kar Tanesi Şema Modeli,

olmak üzere iki yöntem mevcuttur.

3.5.1 Yıldız Şema (Star Schema)

Yıldız şema Şekil 3.13'de gösterildiği gibi merkezde bir tablo ve etrafını saran çok boyutlu tablolardan oluşur. Veri tabanında bir tane olay tablo ve her bir boyut için bir tane boyut tablo yer alır. Yıldız şemada her bir boyut tablosu olay tablosu ile doğrudan ilişkilidir. Boyut tabloları olay tablosu etrafında yıldız şeklinde sıralandığı için yıldız şema olarak adlandırılmıştır. Her boyut tablosunda, boyuta ait nitelikleri tutan kolonlar yer alır. Olay tablosundaki her çoklu her bir boyut tablosunu referans veren göstergeler içerir. Yıldız şemanın etrafında bulunan bu tablolar var olan spesifik iş süreçlerini göstermekte ve sayısal veriler hakkında bilgi sunmak amacıyla kullanılırlar. Verinin çekilmesinde, gösteriminde ve analiz aşamasında kolaylık sağlayan bir yapıdır. Veri ambarı içinde spesifik işlerin olaylarını ve süreçlerini göstermek için kullanılır.



Şekil 3.13 Yıldız Şema

3.5.2 Kar Tanesi Şema (Snowflake Schema)

Kar tanesi şeması yıldız şemasının boyut hiyerarşilerinin birçok boyut tablosunda saklandığı varyasyonudur. Şekil 3.14’te verildiği gibi kar tanesi şeması verileri farklı tablolara böler ve böylece yıldız şemanın sahip olduğu normalizasyon imkânından daha fazlasını sunar. Sadece tek bir tablo bilgi tablosu ile ilişkilendirilir, diğer boyut tabloları da ikincil anahtarları ile bu ilişkiye dâhil olurlar. Bir boyut tanımlamak için birden çok boyut tablosu kullanılabilir.

Yıldız şema, bir olay tablosu etrafında toplanmış boyut tablolarından oluşur ve bu boyutlar denormalize bir yapıdadır. Buna karşılık kar tanesi şema, bir olay tabloya bağlı boyut ve bu boyutlara bağlı başka boyutlardan oluşan bir yapıdır. Kar tane şemada boyutların ikincil anahtarları (foreign key) ile başka boyutlara hiyerarşik bir şekilde bağlanmasının sebebi normalizasyonu sağlamak içindir.

3.6.1 MDX

İlişkisel veritabanlarında tutulan verilere ulaşmak için nasıl SQL (Structured Query Language) dili kullanılıyorsa, MDX(Multi-Dimensional Expressions) de çok boyutlu veritabanlarında tutulan verilere ulaşmak için kullanılan bir sorgulama dilidir. Daha açık olmak gerekirse, MDX Analysis Services ile oluşturulan OLAP veritabanlarını sorgulama için kullanılır ve iki farklı modu destekler. Matematiksel bir ifade olarak kullanıldığında çok boyutlu ifadeleri tanımlamayaabilir veya değiştirebilir aynı zamanda verilerin değerlerini hesaplayabilir. Sorgulama dili olarak kullanıldığında Analysis Services veritabanlarından veri almak için kullanılır.

MDX ile çok boyutlu küplerde saklanan verilerin iki boyut üzerinde sorgulanmasını sağlayacak ifadeler üretilir. MDX kullanılarak çok boyutlu küpler, boyutlar, seviyeler yaratılabilir, mevcut küp yapıları güncellenebilir, küplere veriler eklenebilir ve mevcut veriler sorgulanabilir.

4. TEZ KAPSAMINDA KULLANILAN ÜRÜNLER

4.1 Giriş

Şirketlerin yapmış oldukları işin performans değerini ölçmek için veri analizine ihtiyaç duyarlar. Bu nedenle firmaların en büyük ihtiyaçlarından bir tanesi raporlamalardır. Yazılım projelerinde rapor modülleri büyük önem taşımakta ve yer kaplamaktadır. Şirketler veri analizi yaparak kar ve zarar değerlerini, birim maliyetlerini hesaplayabilirler.

Veritabanının bu kadar önemli olduğu bir ortamda şüphesiz verilerin analizi ve şirketlerin organizasyonları içerisindeki bir takım stratejik kararlarını, şirket geçmişi ile ilgili doğru saptamalarını bu veritabanı aracılığı ile yapabiliyor olması önem taşımaktadır.

Birçok organizasyon ilişkisel ve OLAP veritabanı motoru geliştirmiştir. Bununla birlikte birçok organizasyon da OLAP veritabanı motoru ile etkileşim kurabilen arayüzler geliştirmiştir.

Bütün bu bilgiler göz önünde bulundurularak tez kapsamında, veri ambarı ve OLAP teknolojileri kullanılarak çekilen anlık raporlamaların karar destek mekanizmasını oluşturan birçok verinin günlük aylık, yıllık olarak incelenmesi sağlamaktır.

Sorgulama ve raporlama aracının gerçekleştirimi öncesinde bu aracı besleyen verilerin oluşturulması ve çok boyutlu veri yapılarında saklanması gerekmektedir. Bu kapsamda ilişkisel veri tabanı ve veri ambarı amaçlı kullanılan OLTP yapıları, OLAP için kullanılan ürün ve uygulama yazılımı geliştirme aracına ait temel bilgiler aşağıdaki bölümlerde yer almaktadır.

4.2 Microsoft Visual Studio .NET 2008

Microsoft firması tarafından sunulan, Windows uygulamaları geliştirilmesini sağlayan yazılım platformudur. .NET çatısı, yazılım geliştiriciler için zengin Windows Forms uygulamalarının geliştirilebilmesi için temiz, nesneye dayalı ve genişletilebilir sınıf kümelerini sunar. Uygulamalar, çok katmanlı dağıtık çözümlerde yerel kullanıcı arayüzü gibi davranır.

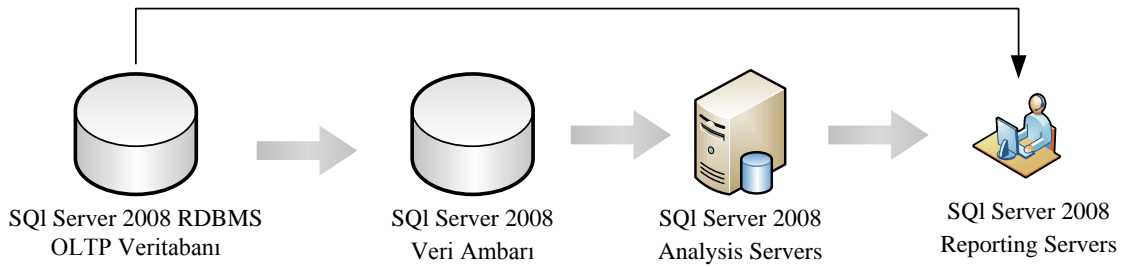
4.3 Microsoft Sql Server 2008

Microsoft SQL Server 2008 Veritabanı Yönetim Sistemi, ilişkisel veri tabanı sistemidir. Enterprise ve Development sürümleri mevcuttur. Verilerin tutulduğu iki boyutlu tablolar, belirlenmiş olan kısıtlara bağlı kalarak diğer tablolar ile kurulmuş ilişkiler üzerinden veri bütünlüğünü ve doğruluğunu sağlamaktadır. Veri tabanı, sistem, kullanıcı ve kontrol bilgilerinin tutulduğu fiziksel ve mantıksal yapıları içerir.

Tez kapsamında oluşturulan veri ambarı, SQL Server 2008 Development sürümü üzerinde yer almaktadır. Veri ambarının oluşturulmasında kullanılan veri dönüştürüm servisi, SQL Server 2008 ürününün bir parçasıdır.

4.4 Microsoft Analysis Services 2008

Microsoft firması tarafından sunulan, Windows uygulamaları geliştirilmesini sağlayan yazılım platformudur. .NET çatısı, yazılım geliştiriciler için zengin Windows Forms uygulamalarının geliştirilebilmesi için temiz, nesneye dayalı ve genişletilebilir sınıf kümelerini sunar. Uygulamalar, çok katmanlı dağıtık çözümlerde yerel kullanıcı



Şekil 4.1 Analysis Services

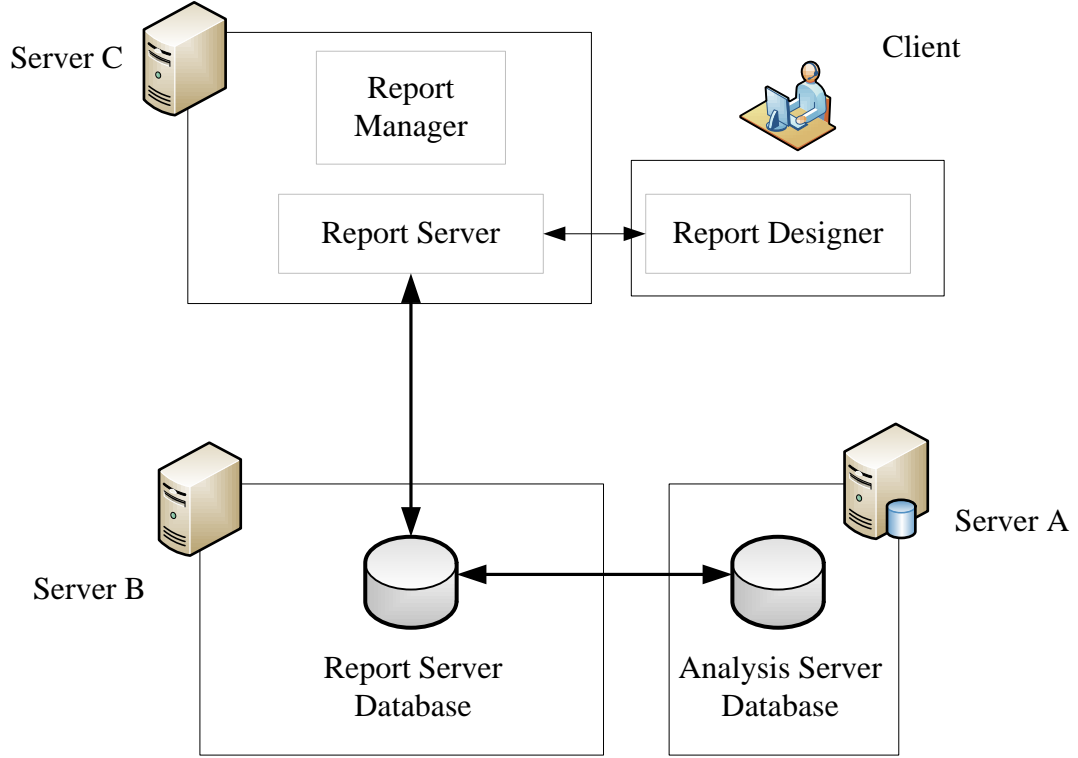
SQL Server 2008'in en önemli servislerinden biri olan SQL Server Analysis Services, karar destek motorunun ve araçlarının yer aldığı ortamdır. Karar destek mekanizmasına ait iki içerik olan veri madenciliği ve OLAP bu ürün kapsamında desteklenmektedir.

Analysis Services mimarisi istemci ve sunucu bölümleri olmak üzere ikiye bölünebilir. İstemci bölümü, uç kullanıcılar için arayüz desteğini sağlarken sunucu bölümü, istemci servislere işlevsellik ve güç veren motorların çalışmasını sağlar. OLTP ile doğrudan bağlantı kurarak çalışabildiği gibi birçok veri tabanı sistemine ODBC üzerinden bağlanabilmektedir.

4.5 Microsoft Reporting Services 2008

SQL Server Reporting Services (Raporlama Servisleri), SQL 2000 ile birlikte bir ek olarak sunulmuş olup SQL Server veya OLE DB ve ODBC tabanlı farklı veri kaynakları üzerinde raporlama yapmak için kullanılır. SQL 2008 ile birlikte daha da olgunlaşan bu ürün piyasadaki diğer araçlara göre yapılandırılması ve kullanımı kolay olan bir araçtır. Reporting Services, SQL Server 2008'in bir parçası haline getirilmiş olup SQL Server içerisinde bir servis olarak sunulur. Bu aracı kullanarak veri kaynaklarındaki sorguların sonuçları XML, CSV, TIFF, EXCEL, PDF, Tek dosyalı Web sayfası(Multipart MIME Message-MHTML) formatında dışarı verilebilir. Bu aracın en güzel yanı Microsoft tabanlı birçok ürünle ilişkili çalışıyor olması, SOAP ve

WMI arabirimlerini destekliyor olması ve .NET uygulamalarında gömülü olarak kullanılıyor olmasıdır.



Şekil 4.2 Reporting Services

Şekil 4.2 'de Server C, web uygulamasının bulunduran sunucu, Server B Reporting Services ile ilgili veritabanının bulunduğu sunucu, Server A Analysis Services ile ilgili veritabanının bulunduğu sunucu ve Client ise raporların Microsoft Visual Studio .NET kullanarak tasarlandığı ortamdır.

Client raporları tasarlar ve Server C'ye deploy eder. Deploy ettiği sırada bu rapor Server B'deki Report Server Database'ine yazılır. Report Server Database'i raporların tüm ayarlarını ve özelliklerini barındırır.

5. VERİ AMBARI VE OLAP TEKNOLOJİLERİNDEN YARARLANILARAK RAPORLAMA ARACI GERÇEKLEŞTİRİMİ

Veri ambarı bir ya da daha fazla işletimsel ya da dış kaynaklı veri kaynaklarından gelen verilerin temizlenerek özetlendiği depo alanıdır. Dolayısıyla, veri ambarı tasarımı öncesinde, veri ambarına veri akışını sağlayacak kaynakların tespit edilmesi gerekir. Tez kapsamında yapılan araştırma sonucu edinilen bilgi birikimi ile işletimsel bir sistem olan ve ilişkisel veritabanı üzerinde verilerini saklayan Pamukkale Üniversite Hastanesi'nde kullanılan Hastane Bilgi Yönetim Sistemi (HBYS) veri kaynağı olarak seçilmiştir. Bu sistem hastanın hastaneye gelişinden itibaren gerçekleşen her türlü muayene, tetkik, tıbbi işlem, ameliyat, reçete ve sarf, satılabilir malzeme hareketleri gibi tüm işlemlerini kapsamaktadır. Bu sisteme entegre olarak hazırlanan Malzeme Yönetim Sistemi (MYS), malzeme ve ilaçların hastaneye ilk girişinden itibaren hastane içerisinde hareketlerini tutmak için tasarlanmıştır. Satılabilir ve sarf malzeme istek ve takipleri taşınabilir yönetmeliğine göre belirlenmiştir. Bu veriler Microsoft SQL Server 2005 veritabanı yönetim sistemi üzerinde yer almaktadır

HBYS ve MYS veritabanlarında yer alan yüksek hacimli veri üzerinde hazırlanan statik raporlar genel ihtiyaçlar düşünülerek hazırlanmıştır. Kullanılan sistem kullanıcılara anlık ihtiyaçlarına hızlı cevap verecek esnek bir araç ile istedikleri noktalarda kırılımlar ve toplamlar olarak analiz yapabilecekleri bir araç sunmak amacıyla bu projeye başlanmıştır. Bu araç sayesinde rapor geliştirme yükünün sistem yöneticilerinin üzerinden alınacağı ve mevcut raporların bakımının kolaylaşacağı düşünülmüştür.

Bu uygulama öncesinde veri ambarı ve OLAP teknolojisini bir arada kullanarak organizasyonlara esnek bir raporlama ve analiz platformu sunan ürünler incelenmiştir. Bu ürünleri satın almadan önce getireceği faydaları daha iyi görebilmek adına, üniversitenin elinde bulunan mevcut yazılım lisansları ile sunucu tarafında Microsoft

SQL Server 2008, istemci tarafında ise Microsoft Reporting Service kullanılması kararlaştırılmıştır.

Geliştirilen uygulamanın belirli bir süre kullanımından sonra kullanıcıların bu tür bir ürünün yetenekleri ile tanışması, mevcut işlerinde kullanabilme becerilerinin artması ve işletme içinde genel olarak iş akışlarına olumlu bir katkı yapması hedeflenmiştir. Microsoft SQL Server 2008 üzerinde gelen Analiz Hizmetleri kullanılarak OLAP küpleri hazırlanmış ve kullanıcıların Microsoft Reporting Service’de hazırlanan raporlamalarla bu küplere ulaşımı sağlanarak çok boyutlu veri analizi imkânı sağlanmıştır. Bu kısımda HBYS işlem sisteminin yapısı ve buna entegre olarak çalışan OLAP uygulamasının nasıl geliştirildiği anlatılacaktır.

5.1 Hastane Bilgi Yönetim Sistemi Yapısı

Hastane Bilgi Yönetim Sistemi iç içe geçmiş ve paralel olarak çalışan modüllerden oluşmaktadır. En önemlileri arasındakiler aşağıda verilmiştir;

- Hasta Kayıt-Arşiv Modülü
- Poliklinik Yönetim Modülü
- Eczane Modülü
- Malzeme, Stok Takip ve Demirbaş Modülü
- Laboratuvar Modülü
- Hasta Yatış-Çıkış Modülü
- Fatura ve Muhasebe Modülü

Pamukkale Üniversite Hastanesi’nde kullanılan HBYS programı 2000 yılından beri geliştirilmesi yapılmaktadır. Sistemde veriler Microsoft Sql Server 2005 üzerinde tutulmaktadır. Hasta kayıtlarının tutulması, hizmetlerin faturalanması, gelir-gider takibinin yapılması, kaçakların önlenmesi, hizmet kayıplarının engellenmesi, hastane kaynaklarının en verimli şekilde kullanılması, stok takibinin yapılması ve hastanenin sistemlerini içermesi HBYS’nin gereklerindedir. Sistemin gün içerisinde yapılan bu binlerce işlemi kısa sürede gerçekleştirebilmesi beklenmektedir.

Raporlamaya yönelik sorgulamalar HBYS ve MYS üzerinden gerçekleştirilir. Bu sistemler Delphi 7 programla dili kullanılarak hazırlanmıştır. Raporlama sistemi olarak Delphi 7 ile uyumlu olarak çalışan Report Builder yazılımı kullanılmaktadır. Sistemin zamanla ortaya çıkan ihtiyaçlar ışığında büyümesi, kanunlarda yapılan değişiklikler ve veri hacminin artmasıyla birlikte dinamik raporlama ve çok boyutlu veri analizi ihtiyacı artmıştır.

Bu kadar hızlı büyüyen bir sistemde özel isteklere bağlı olarak istenen istatistiksel raporların hazırlanması oldukça karmaşık hale gelmiştir. Karmaşıklıkların hem verilere ulaşmak için hazırlanan sorguların tasarlanması hem de çekilmesi esnasında ortaya çıktığı görülmektedir. Veritabanında kullanılan normalizasyon nedeniyle detay seviyesi yüksek verilerin tutulması ve verilerin yoğunluğu istenilen istatistiksel verilerin elde edilmesini zorlaştırmaktadır. Hem sorgunun karmaşıklığından hem de verilerin çokluğundan dolayı üretilen sorguların cevaplanma süresi oldukça uzun zaman almaktadır. Sistemden çekilen sorguların detay seviyelerinin artması ve sorgulamaların birden fazla tabloyu etkilemesi sorgulama sırasında diğer kullanıcıların yapacakları işlemler üzerinde yavaşlamalara neden olacaktır.

Bu problemlerin veri ambarı ve OLAP yöntemleri ile aşılabileceği düşünülerek tek kapsamında bu teknolojiler üzerinde durulmuştur.

Bu projeye konu olan çalışmada aşağıdaki adımlar izlenmiştir;

- Microsoft Sql Server ilişkisel veritabanı üzerinde veri ambarı için kullanılacak gerekli tablo tanımlarının yapılması,
- HBYS veritabanında kullanılan verilerin temizlenerek Microsoft Sql Server 2008 veritabanında tasarlanan veri ambarına taşınması,
- Microsoft Sql Server Analysis Services kullanılarak OLAP küplerinin oluşturulması ve veri ambarındaki verilerin analiz sunucu üzerindeki OLAP küplerine aktarılması,
- Microsoft Sql Server Reporting Services kullanılarak OLAP küplerinin sorgulanması.

OLAP veritabanı oluşturulurken hastane içerisinde kullanılan maliyet raporları göz önünde bulundurulmuştur. Maliyet raporlarında kullanılan sorguların diğer

sorgulamalardan farkı bu tür sorgulamaların yavaşlığı, ilişkisel veritabanındaki tablolardan hareket tablolarını kapsaması ve sorguların çalıştırılması aşamasında diğer işlemlerin yavaşlamasıdır. Bütün bu olumsuzların en aza indirmesi için küp ve boyut tabloları hazırlanmıştır.

5.2 Veri Ambarı Tasarımı

Veri ambarı için oluşturulan ilişkisel veri tabanı Microsoft SQL Server 2008 veritabanı üzerindedir. Veri ambarında bulunan tablolara ait varlık bağıntı şekilleri Şekil 5.1 – Şekil 5.5 ile gösterilmiştir. Tablolar iki kısımda yapılandırılmıştır. "Dim" şeması kullanılan tablolar boyut tablolarını, "Fact" şeması kullanılan tablolar olay tablolarını ifade etmektedir. Olay tablolar, sayılabilir nitelikleri içeren tablolardır ve boyut tabloları ile ikincil anahtarlar üzerinden ilişkilidirler. İkincil anahtarlar yıldız şemanın etrafında bulunan boyut tablolarının içindeki birincil anahtarlardır. Bu anahtarların kombinasyonu her bilgi tablosu kaydı için tekil niteliktedir. Veri ambarında bulunan ve istatistiksel sorgular hazırlamak amacıyla gerekli verileri tutan tablo tanımları aşağıda verilmiştir.

5.2.1 Veri Ambarı Olay (Fact) Tabloları

Olay tabloları, analiz etmek istenen anahtar iş parametrelerini temsil eder. Olay tabloları, hesaplamaların temellerini oluşturur ve iş bilgisini içeren sayısal veriyi sunan yıldız şemanın merkezinde bulunur. Aşağıda bu projede kullanılan olay tabloları verilmiştir.

- **Acilİslemler:** Acil Serviste yapılan işlemleri acil işlem türleri ve tarihe göre tutan olay tablosudur.
- **Ameliyatlar:** Yapılan ameliyat işlem sayılarını ameliyat türü, anestezi ve hasta durumu, bölüm ve tarihe göre tutan olay tablosudur.
- **BirimeCikilanMalzemeler:** Hastane içerisinde bulunan birimlere çıkılan sarf malzemelerin sayısını ve fiyatlarını malzeme türü ve tarihe göre tutan olay tablosudur.

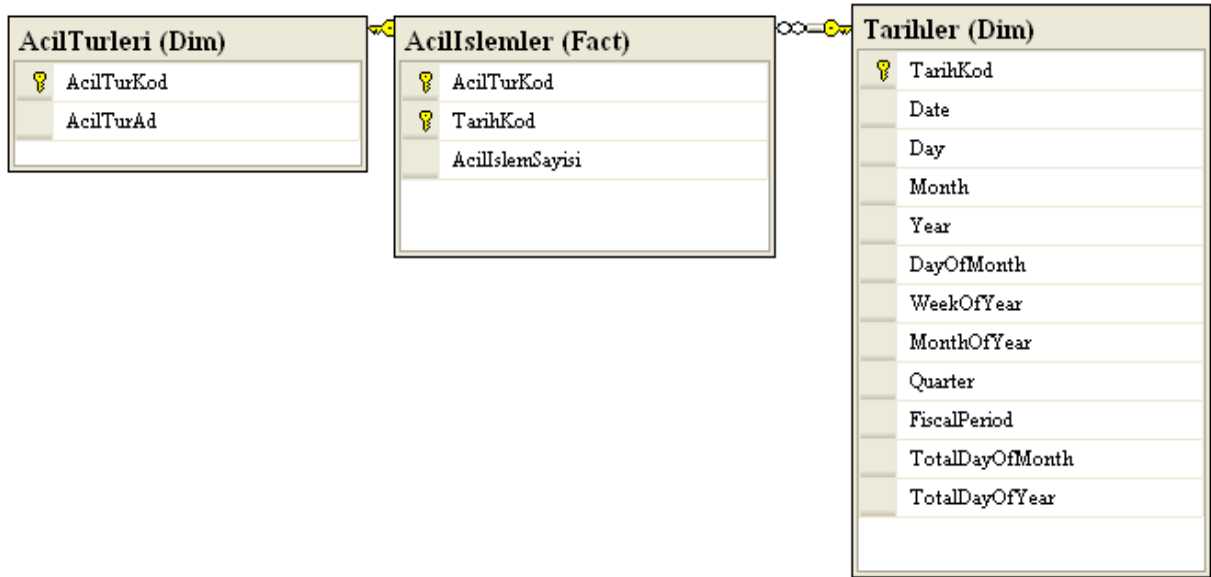
- DogrudanKatkiPayi: Öğretim üyelerini yapmış oldukları işlemleri sayılarını hasta, işlem yapılan bölüm, uzmanlık ve tarihe göre tutan olay tablosudur.
- Dogumlar: Doğum sayılarını doğum türlerine ve tarihe göre tutan olay tablosudur.
- DolaylıKatkiPayi: Bölüm bazında işlem sayılarını ve yapılan işlemlerin performans miktarlarını tutan olay tablosudur.
- HastaHizmetleri: Hastalara uygulanmış tetkik ve işlem sayılarının tutulduğu olay tablosudur.
- HastayaCikilanMalzemeler: Hastalara kullanılan ilaç, malzeme miktarlarını, alış ve satış toplam fiyatlarını hastaların durumlarına, işlem yapılan servise ve tarihe göre tutan olay tablosudur.
- LabIslemleri: Yapılan laboratuvar sayılarını tetkik türlerine ve tarihe göre tutulan olay tablosudur.
- Muayeneler: Muayene işlem sayılarının öğretim üyesi, yapılan bölüm ve tarihe göre tutulduğu olay tablosudur.
- Patolojiler: Patoloji işlem sayılarını patoloji tür, hasta durum ve tarihe göre tutan olay tablosudur.
- Radyolojiler: Radyoloji işlem sayılarının tetkik tür, hasta durum ve tarihe göre tutan olay tablosudur.
- Receteler: Tarih boyutuna göre reçete sayılarının tutulduğu olay tablosudur.
- YatisiYapilanlar: Uzmanlık dallarına ve tarih boyutuna göre yatışı yapılan hasta sayılarının tutulduğu olay tablosudur.

5.2.2 Veri Ambarı Boyut (Dimension) Tabloları

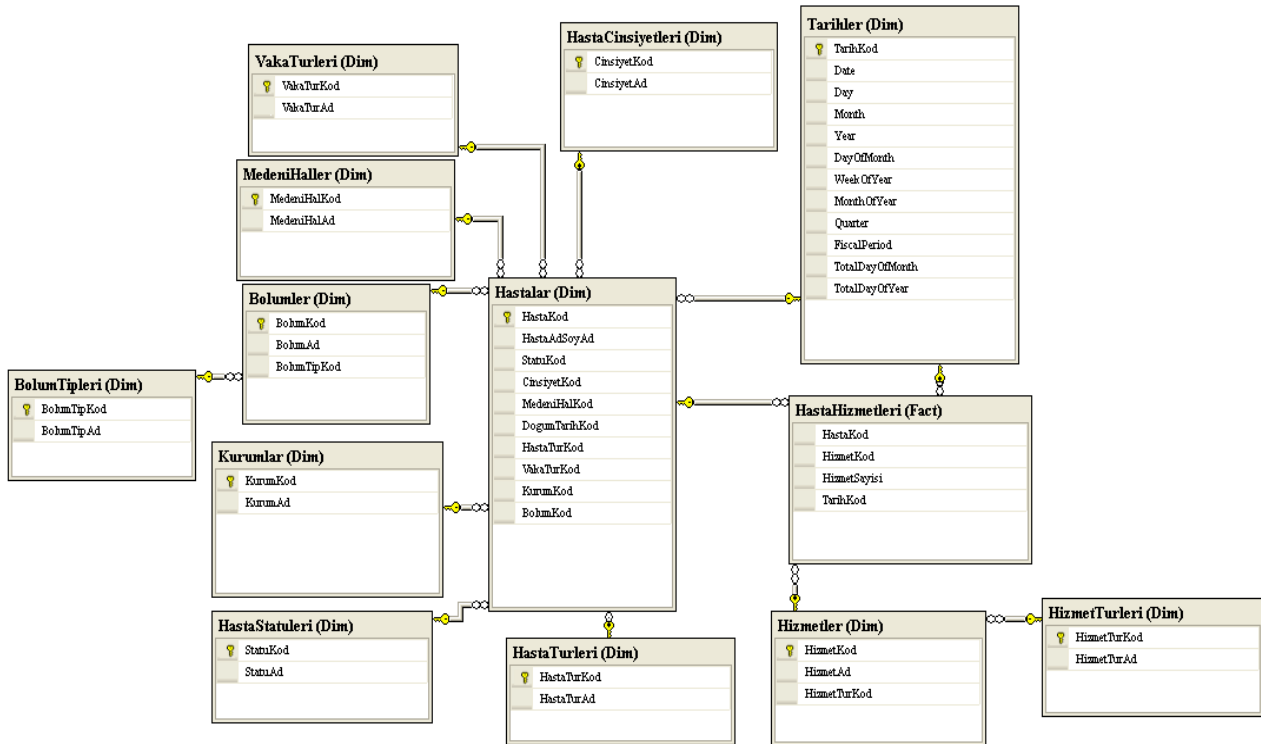
Boyut tabloları, olay tabloları için bir alan oluşturur, sonucu etkileyen eksenlerdir. Toplanan, ortalaması alınan vb. olaylar tarafından işin görünümü tanımlanır. Boyut tablolarının oluşturulmasının amacı kullanıcının ihtiyacı olan kullanılabilir, tanımlayıcı

analitik bilgileri kullanıcıya anlaşılır bir dille sunmaktır. Boyut tabloları yıldız şema içerisinde sadece tek bir iş kaydını sunarlar. Proje kapsamında olay tablolarına bağlı boyut tabloları aşağıda verilmiştir.

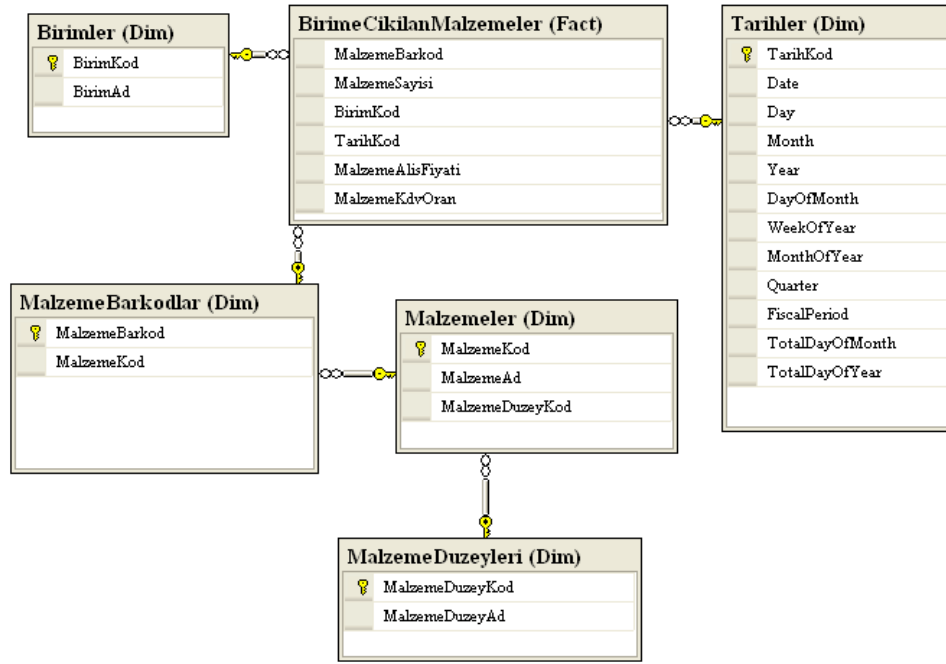
- AcilTurleri: Acil işlem türlerini tutan boyut tablosudur.
- AmeliyatTurleri: Ameliyat türlerini tutan boyut tablosudur.
- AnesteziDurumlari: Yapılan ameliyatlarda anestezi için gerekli bilgileri tutan boyut tablosudur.
- Birimler: Hastane içerisinde bulunan birimleri tutan boyut tablosudur.
- Bolumler: Hastane içerisindeki polikliniklerin tutulduğu boyut tablosudur.
- Doktorlar: Doktor bilgilerinin tutulduğu boyut tablosudur.
- HastaDurumlari: Hastaların durumlarının tutulduğu boyut tablosudur.
- Hastalar: Hasta bilgilerinin tutulduğu boyut tablosudur.
- HastaTurleri: Hasta türleri ile ilgili bilginin tutulduğu boyut tablosudur.
- Hizmetler: Hizmetler ile ilgili bilgilerin tutulduğu boyut tablosudur.
- Kurumlar: Hastaların kurumları ile ilgili bilgilerin tutulduğu boyut tablosudur.
- LabTetikikleri: Laboratuvar tetkikleri ile ilgili bilgilerin tutulduğu boyut tablosudur.
- Malzemeler: Malzeme ve ilaç bilgilerinin tutulduğu boyut tablosudur.
- PatalojiTurleri: Patoloji türlerini tutan boyut tablosudur.
- RadyolojiTetikikleri: Radyoloji tetkiklerini tutan boyut tablosudur.
- Tarihler: Tarihleri ile ilgili bilgilerin tutulduğu boyut tablosudur.



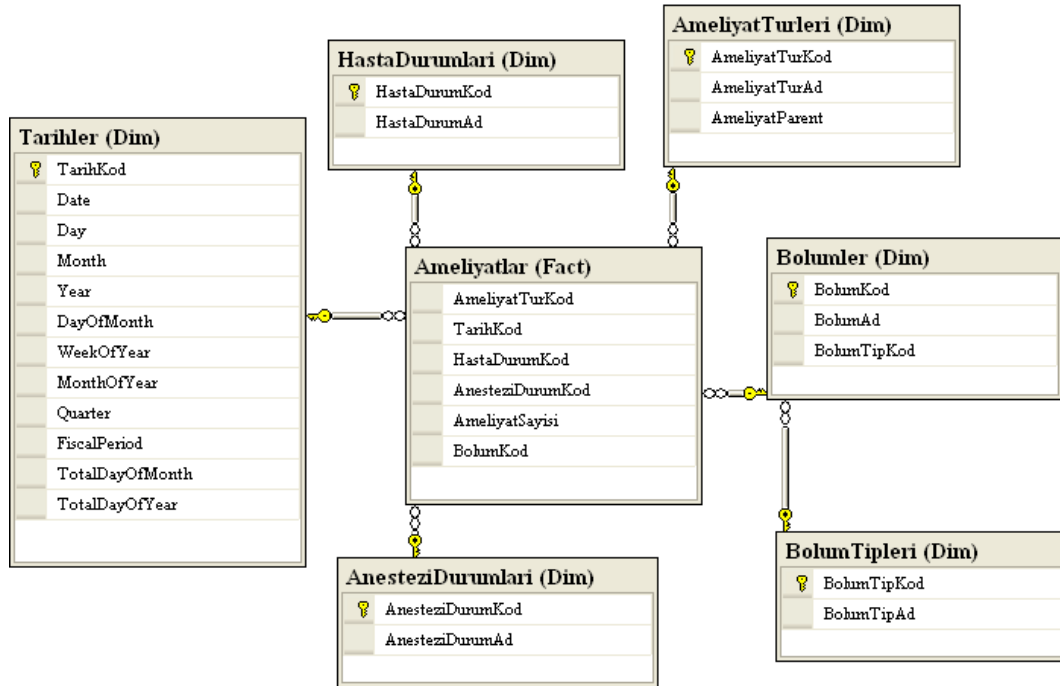
Şekil 5.1 Acil İşlemler ER Diagramı



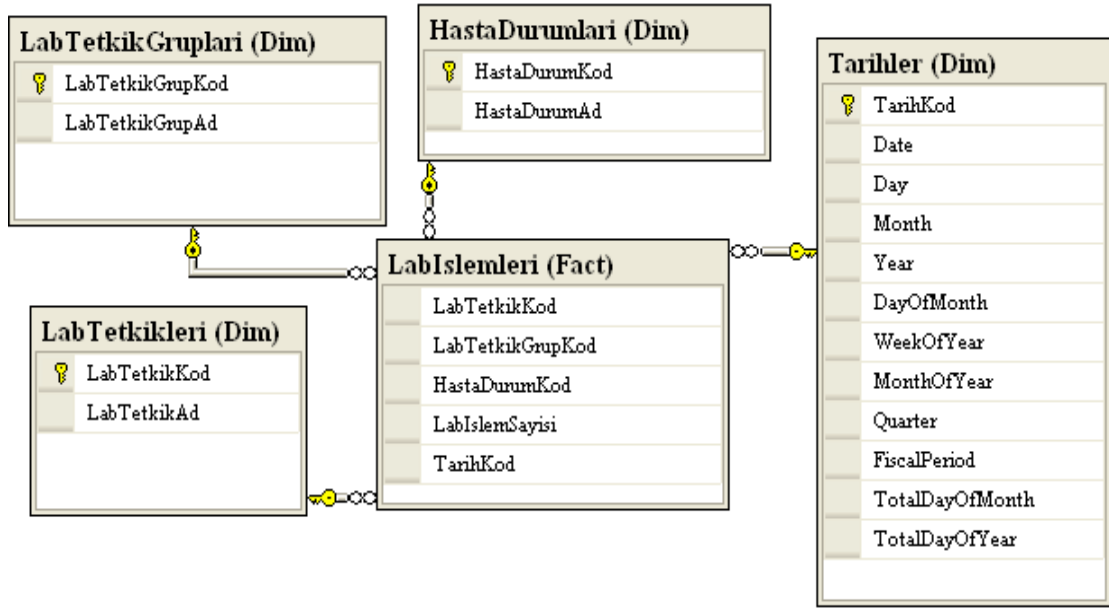
Şekil 5.2 Hasta Hizmetleri ER Diagramı



Şekil 5.3 Birime Çıkılan Malzemeler ER Diagramı



Şekil 5.4 Ameliyat İşlemleri ER Diagramı



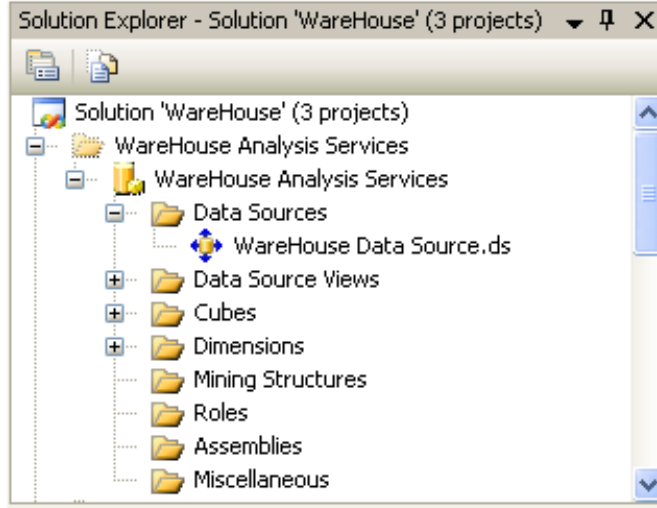
Şekil 5.5 Laboratuvar İşlemleri ER Diagramı

5.3 Veri Ambarı Üzerinde Küp ve Boyutların Tanımlanması

Küpler içerisinde özel olarak hesaplanmış ya da özetlenmiş olay ve çok boyutlu verileri tutar. Verileri de içerecek şekilde oluşturulan veri ambarı üzerinde çok boyutlu küplerin tanımlanması için Microsoft Analysis Services aracı kullanılmıştır. Bu araç ile Microsoft SQL Server üzerinde bulunan veri ambarına erişim sağlanarak, boyut ve olay tabloları seçilerek küp tanımları hazırlanmıştır. Aşağıda tez kapsamında kullanılan küp tanımları verilmiştir.

- Acil İşlemler: Acil servise yapılan hasta giriş sayılarının tarih ve işlem türüne göre gruplayan küp tanımıdır.
- Ameliyathane: Ameliyat sayılarının anestezi durumu, hasta durumu, ameliyat türü ve zaman boyutuna göre gruplayan küp tanımıdır.
- Birim Malzemeleri: Birimlere çıkılan sarf malzemelerinin adı, toplam sayıları, fiyatları ve tarih boyutuna göre tutulduğu küp tanımıdır.
- Doğrudan Katkı Payı: Doktorların yapmış oldukları işlem ve performans sayılarını, bölüm ve hizmet ve tarihe bağlı olarak tutan küp tanımıdır.

- Dolaylı Katkı Payı: Hastalara yapılmış olan işlem ve performans sayılarını, bölüm ve hizmet ve tarihe bağlı olarak tutan küp tanımıdır.
- Hasta Hizmetleri: İleri veri madenciliği uygulamaları için kullanılabilir olan hasta detay bilgilerine göre hastalara yapılmış toplam işlem sayılarını tutan küp tanımıdır.
- Laboratuvar İşlemleri: Laboratuvar işlem sayılarının tetkik türleri ve grupları, hasta durumu ve zaman boyutuna göre gruplayan küp tanımıdır.
- Muayene: Doktor, bölüm, bölüm tipi, hasta durumu ve zaman boyutları ile muayene sayılarını gruplayan küp tanımıdır.
- Patoloji: Patoloji türleri, hasta durumu ve zaman boyutuna göre gerçekleşmiş patoloji sonuçlarının sayılarını veren küp tanımıdır.
- Radyoloji: Radyoloji işlem sayılarının tetkik türü, hasta durumu ve tarih boyutlarıyla ifade edildiği küp tanımıdır.
- Reçeteler: Zaman boyutuna göre verilmiş reçete sayılarının tutulduğu küp tanımıdır.
- Yataklar: Hastanede bulunan yatak sayılarının yatak türüne ve tarih boyutuna göre tutulduğu küp tanımıdır.
- Yatışı Yapılan: Uzmanlık dallarına ve tarih boyutuna göre yatışı yapılan hasta sayılarının tutulduğu küp tanımıdır.
- Malzemeler: Poliklinik, servis ve yoğun bakımlarda kullanılan ilaç ve medikal malzemelerin toplam miktarları, malzeme- ilaç adı ve türü, alış ve satış fiyatları ve tarih boyutuna göre tutulduğu küp tanımıdır.



Şekil 5.6 Analiz Sunucuda Yeni Veri Kaynağı Tanımlanması

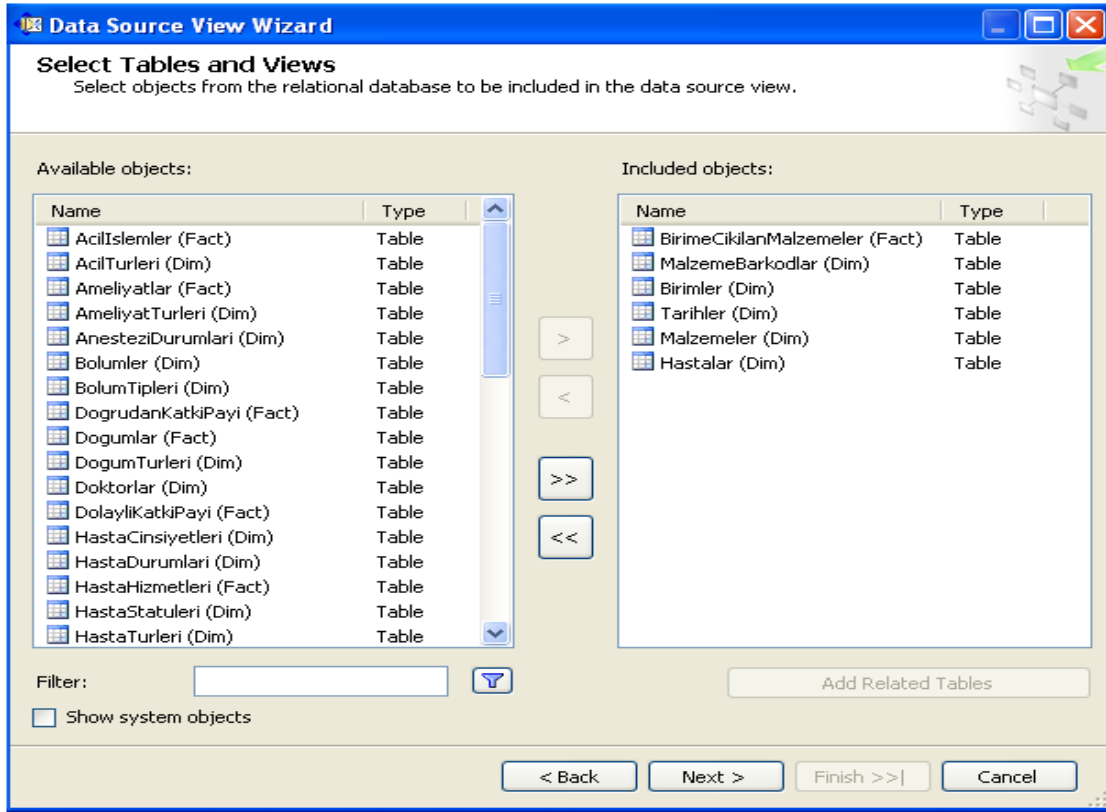
Analiz yöneticisi ekranında bağlanılan analiz sunucusuna açılan menüden yeni bir veritabanı “Data Sources” sekmesinden oluşturulur. Şekil 5.6’de OLAP küp ve boyutlarının oluşturulması sırasında verinin alınacağı kaynaklar “Data Sources” sekmesinde tanımlanır. Kaynak olarak birden fazla veritabanı tanımlanabilir. Oluşturulan connection string “Provider=SQLNCLI10.1;Data Source=.;Integrated Security=SSPI;Initial Catalog=WareHouse” şeklinde olacaktır.

“Provider” bağlantı sağlayıcıların ismini tutar. Bu projede Sql Server 2008 ilişkisel veri tabanı kullanıldığı için “Sql Server Native Client 10.0” seçilmiştir.

“Data Source” sunucu adını için kullanılır. Sql Server için kullanılan sunucu adı “.” verilerek yerel sunucu seçimi yapılmıştır.

“Integrated Security” MS Sql Server sunucusuna Windows hesabı ile bağlantı yapılacağını belirtir. “True”, “False” veya “SSPI” girilebilir. “SSPI”, “True” ile eş anlamlıdır ve bu durumda Windows hesabı kullanılır.

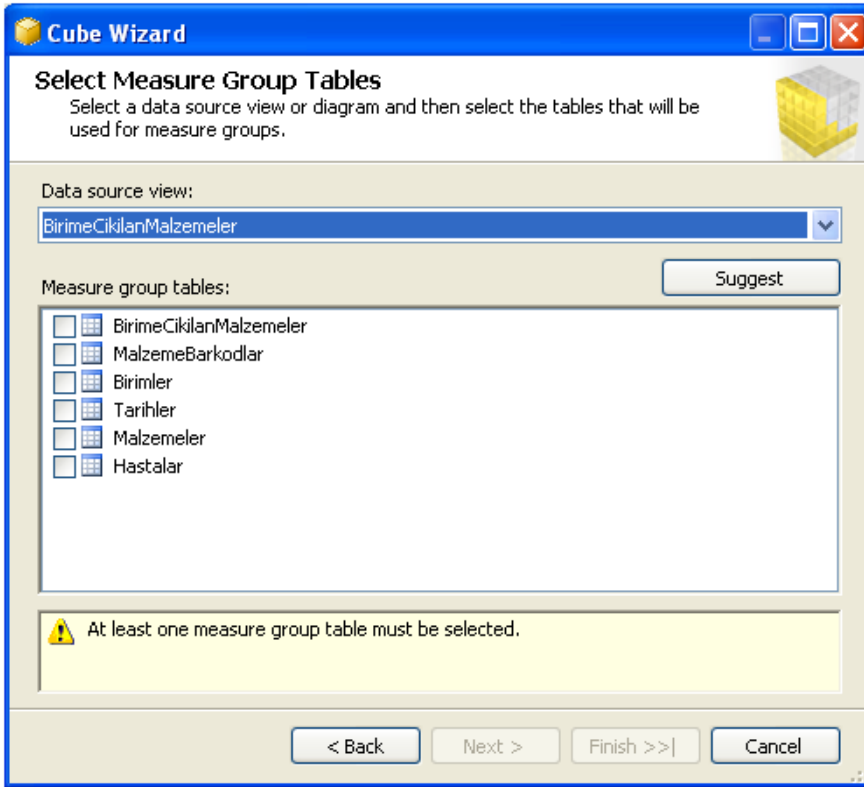
“Initial Catalog” veri tabanı ismidir. Veri tabanı ismi olarak “WareHouse” kullanılmıştır.



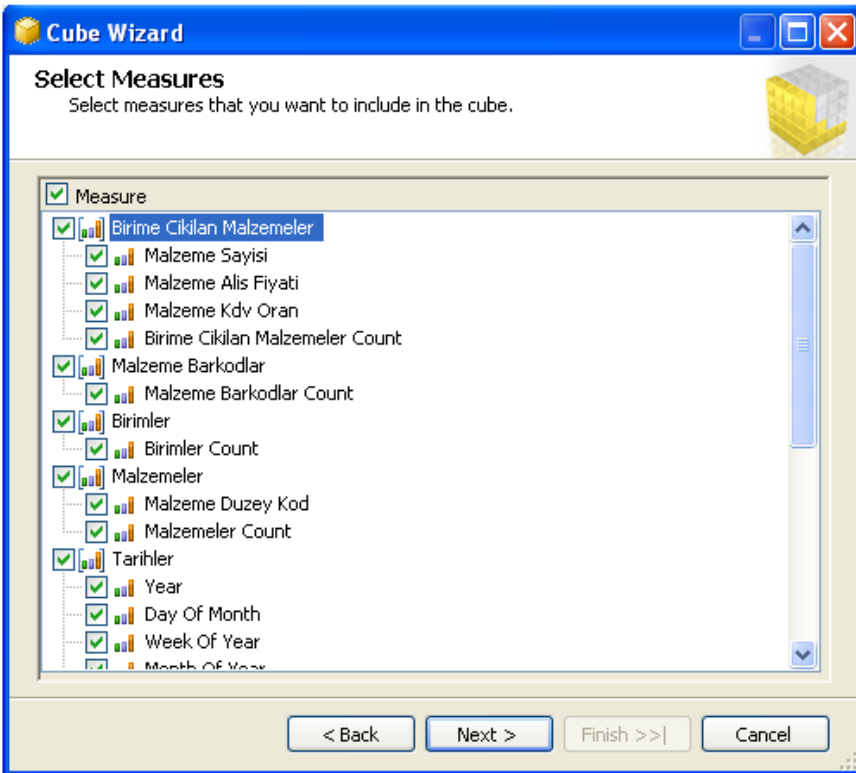
Şekil 5.7 Analiz Sunucuda Yeni Veri Kaynağı Tanımlanması

Boyutların tanımlaması “Data Source View” sekmesi altında yapılmaktadır. Burada sihirbaz yardımıyla boyut yaratılabilmektedir. İlk aşamada yaratılacak boyutun yapısıyla ilgili bir seçim yapılır. Veri kaynağının gösterdiği veritabanı içerisindeki tabloları, viewleri ve aralarındaki ilişkileri içerir. Daha önce anlatılan kar tanesi veya yıldız şema modellerinden uygun olan seçilir. Yıldız şema, tek bir boyut tablosu kullanılarak oluşturulan boyut tanımıdır. Buna karşılık kar yağışı şema, birden çok ilişkili boyut tablosu kullanılarak oluşturulan boyut tanımıdır. Şekil 5.7’de gösterildiği gibi “Yıldız Şema” seçilmiştir. Tanımlanmış veri kaynaklarındaki tabloların listelendiği bu ekranda küp içinde kullanılacak tablonun seçimi yapılır.

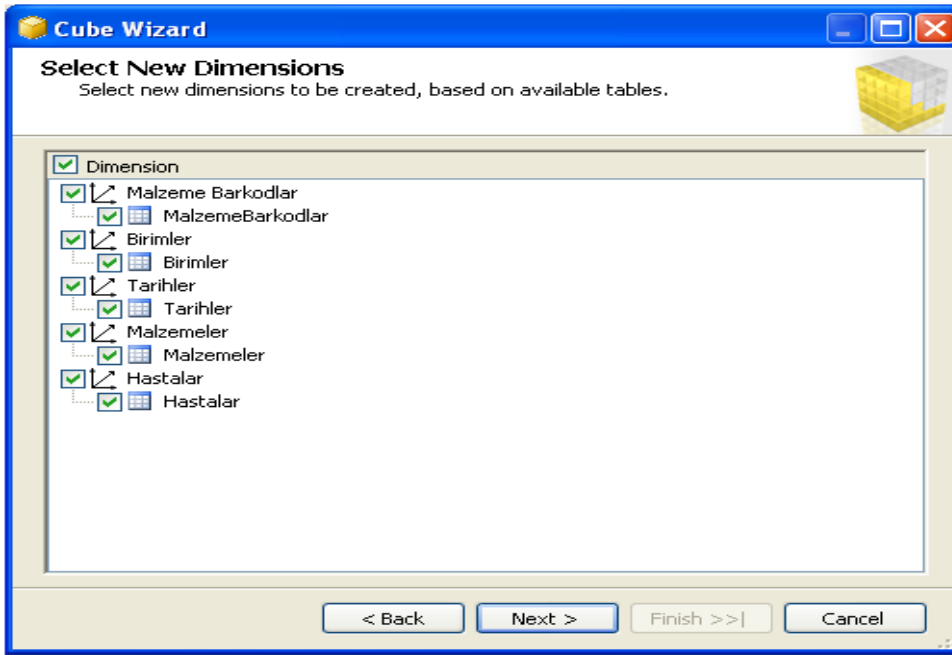
Şekil 5.8’de daha önceden tanımlaması yapılan veri kaynağına göre küp oluşturma ve bu seçim yapılan veri kaynağına göre ölçüm tablo seçimi yapılır. “Data Source View” kısmında daha önceki oluşturduğumuz veriler çıkacaktır.



Şekil 5.8 Küp oluşturma ve Ölçüm Tablo Seçimi



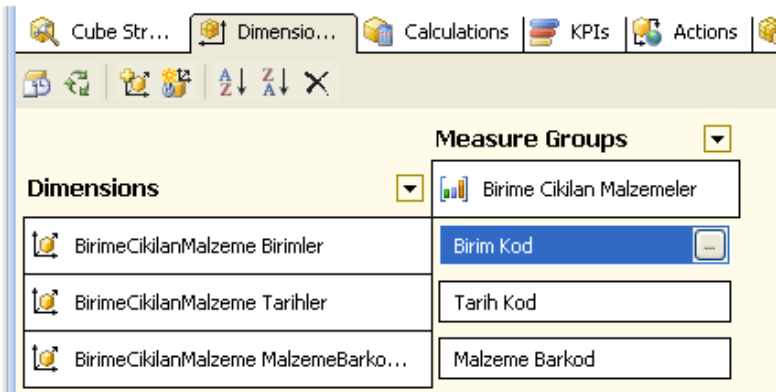
Şekil 5.9 Küp Ölçüm Değerlerinin Belirlenmesi



Şekil 5.10 Küp Boyut Değerlerinin Belirlenmesi

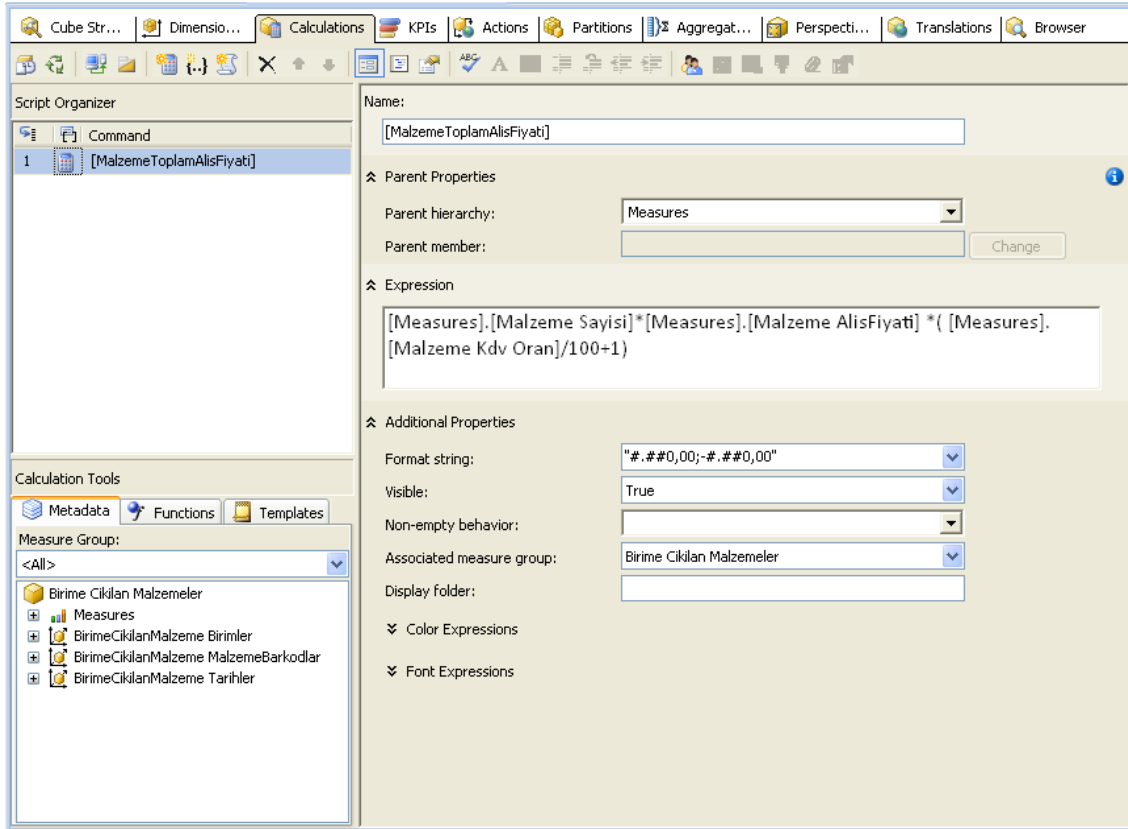
Şekil 5.9’da seçimi yapılan ölçüm tablolarından hesaplanması yapılması istenen değerler seçilir. Küplerle birlikte kullanılacak olan boyutlar tanımlandıktan sonra boyut yaratma kısmına geçmek işleyişi hızlandıracaktır.

Boyutların tanımlanması sonrası içinin doldurulması –verinin çekilerek boyut içine alınması- işlemi gerçekleştirilir. Şekil 5.10’da görüldüğü gibi tablo seçimi sonrası tablo alanları içinden boyut olarak kullanılacak alanlar seçilir.



Şekil 5.11 Dimension Usage

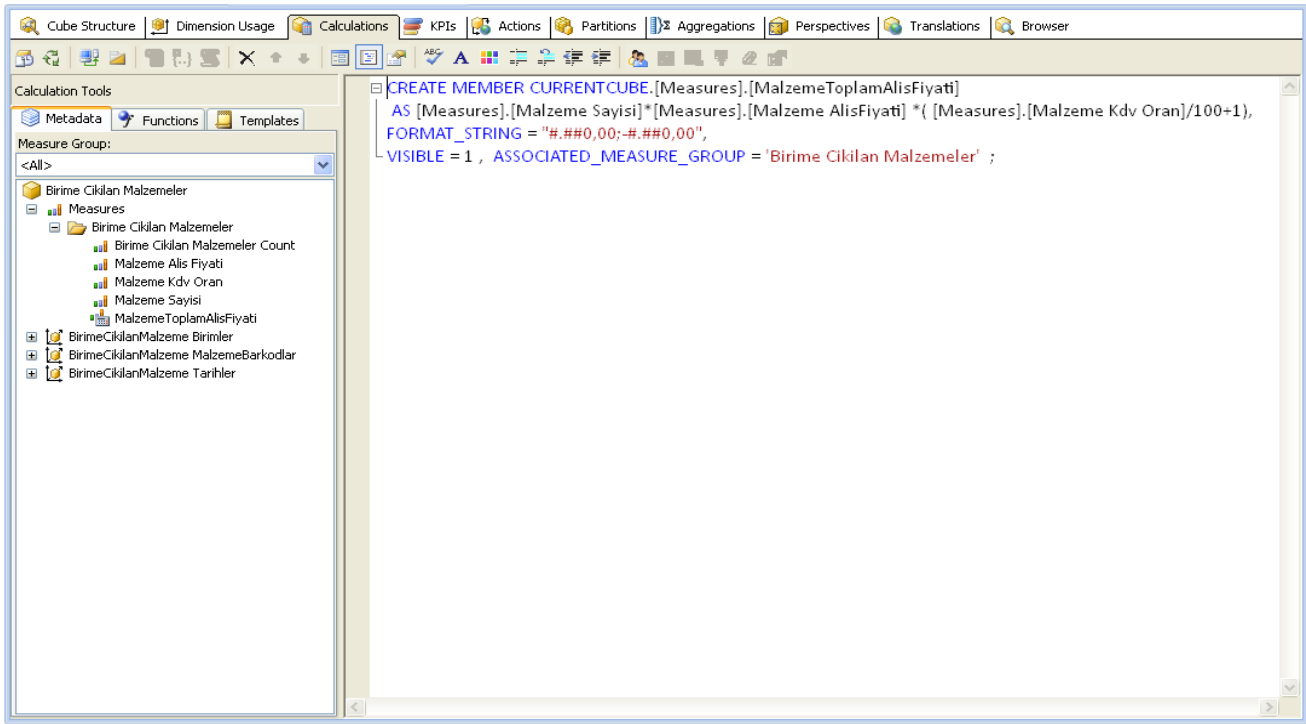
Şekil 5.11’ de verilen Dimension Usage sayesinde daha önceden oluşturulan ölçütler arasında ilişkilerin türlerini belirleyerek (regular, fact, referenced, many-to-many, datamining, no relationship) ilişkiler kurulabilir.



Şekil 5.12 Calculations

Calculations sekmesi sayesinde oluşturulan küp ve kübün alt küplerine ait çeşitli hesaplamalar yapılabilir. Ayrıca var olan değerler üzerinden yeni ölçütler ekleme işlemi de bu bölümde yapılabilmektedir. Yeni bir calculation member tanımlamak için script organizer menüsünden “new calculated member” seçilerek gerekli ayarlamalar yapılabilir.

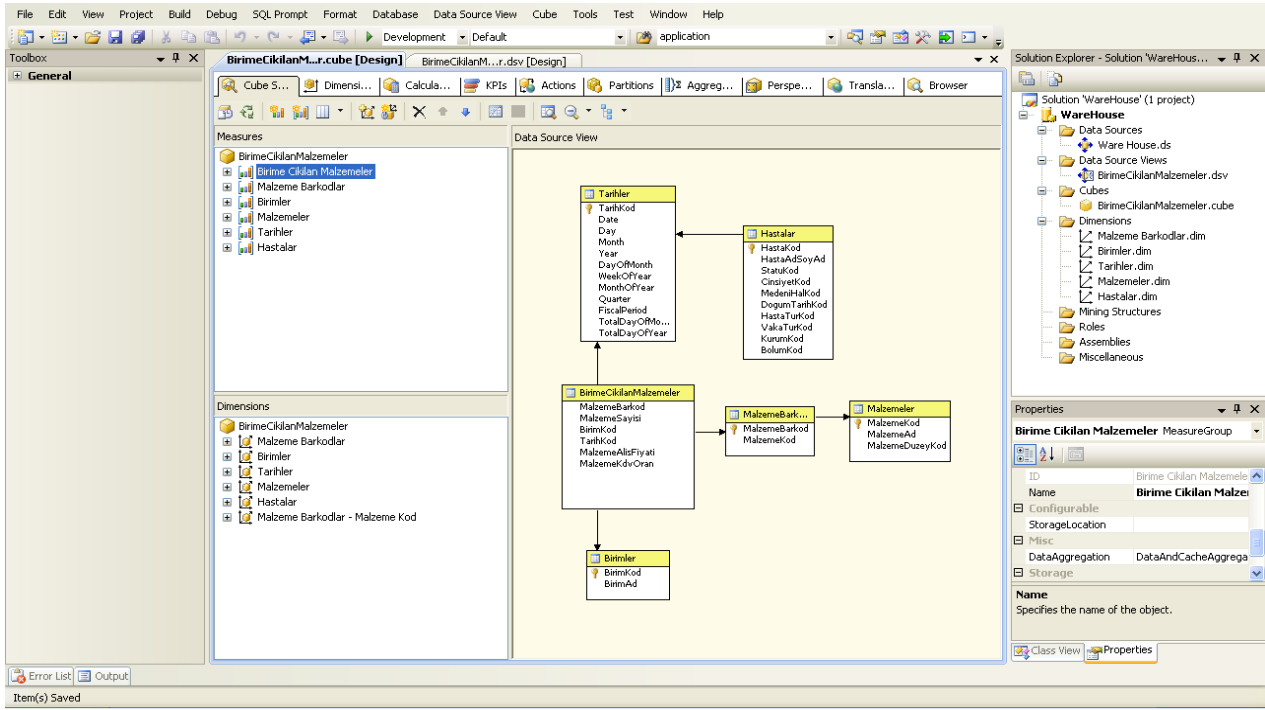
Şekil 5.12’de “Birime Cikilan Malzemeler” kübünde “Malzeme Toplam Alis Fiyati” ölçüm değeri hesaplaması gösterilmiştir. Bu ölçüm değeri oluşturma ile ilgili script örneği Şekil 5.13’te gösterilmektedir.



Şekil 5.13 Calculations Script Örneği

Partitions sekmesinde oluşturulan kübe dair en uygun şekilde sokma ayarlamaları yapılmaktadır. Farklı kesitleri farklı lokasyonlarda barındırmamızı sağlar. Bölüm 3.2’de belirtildiği gibi bir küpün depolanmasında 3 farklı yöntem bulunmaktadır.

Tasarlanan küpler MOLAP modeline göre veri ambarından çekilen veriler küp veri yapıları bünyesine aktarılmıştır. Molap depolama yöntemi detaylı verilerin Analiz Server’da depolanması söz konusu olduğunda tercih edilen yöntemdir. En yüksek performansa sahiptir. Yapılan sorgulamalar ilişkisel veritabanı sisteminde yapmak yerine Analiz Server’da yapılır. Bu yüzden herhangi bir nedenden dolayı ilişkisel veri tabanımızın kapanmasında dahi işlemlerine devam eder.

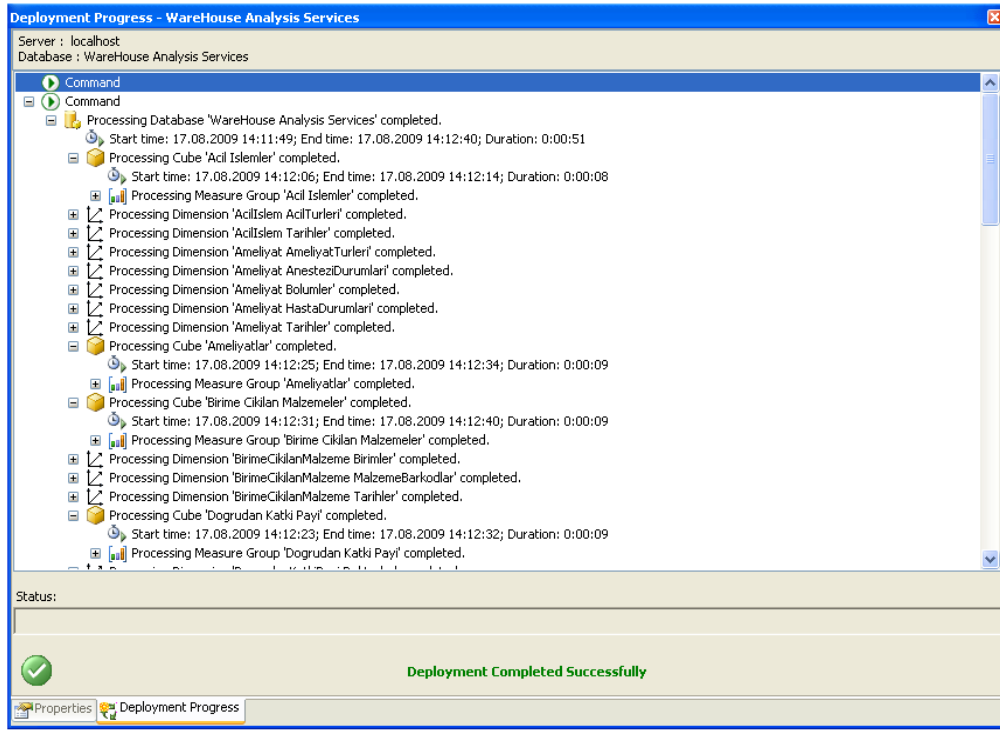


Şekil 5.14 Küp Yapısının Düzenlenmesi

Boyut ve küp tabloları arasındaki bağlantıların görülebildiği Şekil 5.14'deki ekranda son değişiklikler, düzeltmeler yapıldıktan sonra küp kaydedilir ve depolama şekli belirtilerek verinin küp içine alınması işlemi yapılır.

Browser oluşturulan küp içerisindeki verileri göstermeye yarar. Drop Totals or Details Fields yazan bölüme ölçüm değerleri girilir. Aynı şekilde Drop Row Fields kısmına ise oluşturulan boyut değerleri girilir. Oluşacak tabloda yıllara, aylara ve belirtilen aydaki günlere göre veriler kolayca elde edilebilir.

Analysis Services projesi derlendiğinde oluşturulan boyutları ve kübe ait işlemleri yapmaya başlar. Ağaç diyagramı şeklinde sıralanan bu yapı yardımıyla oluşturulan sorgularını görebiliriz. Herhangi bir hata yoksa yükleme işlemi başarı ile tamamlanacaktır.



Şekil 5.15 Küp Yapısının Analiz Sunucuya Yüklenme İşlemi

OLAP veritabanındaki boyut ve küp nesnelere tanımlandıktan sonra tez kapsamında Reporting Services kullanılarak bunlara ulaşım sorgulama veri raporlama işlemi yapılmıştır.

5.4 Veri Ambarını Dolduracak Verilerin Seçilmesi ve Temizlenmesi

Veri ambarı tasarımı tamamlandıktan sonra hazırlanmış olan tabloların verileri içermesi için gerekli olan verinin çekilmesi, dönüştürmesi, yüklenmesi ve güncelliğini koruması için gerekli olan adımlar gerçekleştirilmiştir. ETL işlemi, heterojen kaynaklarda bulunan verilerin çekilerek, gerekli dönüştürmeler, temizlemeler ve özetlemeler yapılarak veri ambarına yüklenmesi işlemidir.

Tez kapsamında SQL Server 2008 içerisinde bulunan SQL Server Integration Services (SSIS) kullanılarak oluşturulan SSIS paketleri aracılığıyla ETL işlemi yönetilmiştir. Elde edilen veriler için aşağıdaki işlemler gerçekleştirilmiştir.

- Veri çekme işlemi öncesinde ilişkisel veri tabanında veri ambarını dolduracak SQL sorguları hazırlanmış ve çalıştırılmıştır. Böylelikle ne kadar veri ve bu verilerin ne kadar zamanda geldiği incelenme fırsatı elde edilmiştir. Özellikle olay tablolarını dolduracak verilerin sorgulanması zaman alacağı için indeksleme işlemi yapılmıştır. Böylelikle verilerin veri ambarına yüklenme işlemi daha da hızlandırılmaya çalışılmıştır.
- “Tarihler” tablosu yeniden düzenlenmiştir. Yapılan işlem tarihleri veri ambarında bu tablo içerisinde bir değere karşılık gelecek şekilde yeniden güncellenmiştir. Bu işlem yapılarak veriler üzerinde sadece gün, ay, yıl boyutlarının dışında hafta, dönem hatta özel günler içinde sorgulamalar yapılabilir. Veri ambarına doldurulacak olan veriler için “01.01.2009” tarihi başlangıç olarak kabul edilmiştir. Bu nedenle bu verinin karşılığı “Tarihler” tablosunda “1” e karşılık gelir. Bu tarihte işlem gören bütün tarih verilerine “1” yazılmıştır.
- Veri ambarına doldurulacak olan verilerden “NULL” veya “Boş” alana sahip olan değerlerin karşılıkları girilmiştir. Eğer bir karşılığı yoksa bu veriler sisteme dâhil edilmemiştir.
- Olay tabloları boyut tablolarında yer alan birincil anahtar alanlarına göre verilerin yeniden kodlanması sağlanmıştır. Örneğin poliklinik kodları olarak verilen “KBB” için yeniden düzenleme yapılmış ve olay tablolarında “22” olarak girilmesi sağlanmıştır.
- Veri ambarında kullanılan boyut, olay tablo ilişkilendirilmesinde kullanılan ikincil anahtar uygulaması nedeniyle birbirine uygun olmayan verilerin sisteme aktarımı yapılmamıştır.
- Açıklama alanı olarak alınan “Cinsiyet, Medeni Hali, Meslek v.b.” sütunlarının aynı veriyi ifade edecek şekilde düzenlenmesi sağlanmıştır. Örneğin cinsiyet için verilen farklı ifadeler bazı kayıtlarda “E” bazılarında “Erkek” olarak bulunmaktaydı. Bu tür veriler için “Hasta Cinsiyetleri” tablosunda “Bay”, “Bayan” olarak ayırım yapılmıştır.
- Yüksek detaya sahip ilişkisel veritabanından, veri ambarında tasarlanmış boyut tablolarına verilerin düzgün bir şekilde aktarılmasından sonra olay

tablo yapısına uygun olacak şekilde gruplanmış özet veriler aktarılmıştır. Örneğin, hastaya çıkılan malzeme sayıları için veri ambarında belirli bir tarihte, hastanın işlem gördüğü poliklinik türünde, malzeme adı ve hasta durumu için hastaya çıkılmış malzeme sayıları tutulmaktadır. Dolayısıyla veriler çekilirken poliklinik türü, tarih, hasta durumu ve malzeme adına göre gruplanmış özetlenmiş sayısal içerikler olarak veri ambarına aktarılmıştır.

```

SELECT
    Barkod [MalzemeBarkod],
    SUM(Miktar) [MalzemeSayisi],
    Hedef,
    HastalslemPolKod ,
    [Belge Tarihi] [TarihKod],
    SUM(Fiyat) [MalzemeAlisFiyati],
    SUM([Satis Fiyati]) [MalzemeSatisFiyati],
    [Kdv Oran] [MalzemeKdvOran],
FROM (SELECT
    d.Barkod AS [Barkod],
    d.mlzhd_miktar AS [Miktar],
    d.mlzhd_fiyat AS [Fiyat],
    HastalslemPolKod = (SELECT p.polad FROM OPENDATASOURCE('SQLOLEDB', 'Data
Source=192.168.10.2\PAUHASTANE; Initial Catalog=Hospital ;Uid=devrim;Pwd=123456;
').Hospital.dbo.th_Pols p WHERE p.PolKod = ( SELECT IstekYapanBirim FROM
OPENDATASOURCE('SQLOLEDB', 'Data Source=192.168.10.2\PAUHASTANE; Initial Catalog=Hospital
;Uid=devrim;Pwd=123456; ').Hospital.dbo.TH_Islem I WHERE I.Key0=f.Key0)),
    mlzhd_kdv_oran AS [Kdv Oran],
    d.mlzhd_satis_fiyat [Satis Fiyati],
    (SELECT [Dim].[Tarihler].[TarihKod] FROM [Dim].[Tarihler] WHERE [Dim].[Tarihler].[Date] =
mlzhf_belgetarihi) AS [Belge Tarihi],
    f.mlzhf_hedef [Hedef]
FROM OPENDATASOURCE('SQLOLEDB', 'Data Source=192.168.10.2\PAUHASTANE; Initial
Catalog=Hospital ;Uid=devrim;Pwd=123456; ').Hospital.dbo.mys_malzeme_harfis f INNER JOIN
OPENDATASOURCE('SQLOLEDB', 'Data Source=192.168.10.2\PAUHASTANE; Initial Catalog=Hospital
;Uid=ugur;Pwd=123456; ').Hospital.dbo.mys_malzeme_hardetay d ON f.mlzhf_islemkod=
d.mlzhd_islemkod
WHERE 1=1
    AND (mlzhf_belgeturu = '9')
    AND (mlzhf_belgetarihi between '01.01.2009' and '01.08.2009')) AS t
GROUP BY Barkod,[Belge Tarihi],[Kdv Oran],[Hedef],HastalslemPolKod,Hedef
ORDER BY [Belge Tarihi]

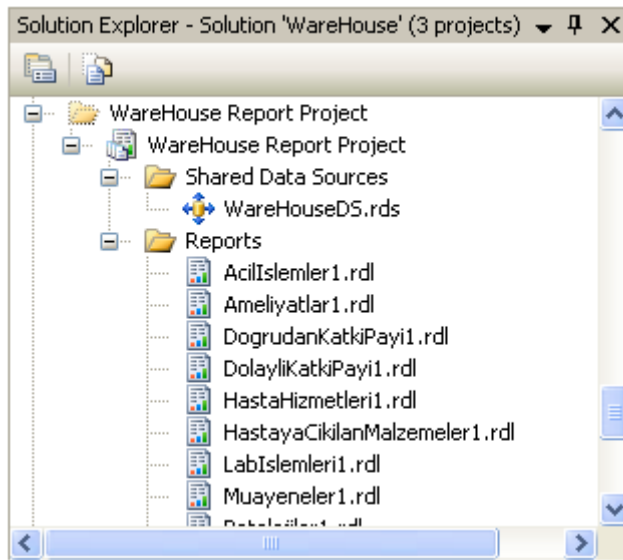
```

Şekil 5.16 “Hastaya Çıkılan Malzemeler” olay tablosunun yüklenmesinde kullanılan SQL dili ile hazırlanmış görev içeriği

Veri dönüştürme servisi ile Şekil 5.16'da gösterildiği gibi tüm boyut ve olay tablolarına ait veri aktarımını sağlayan SQL sorguları hazırlanmış ve bu sorgular işletilerek verilerin veri ambarına aktarımı sağlanmıştır.

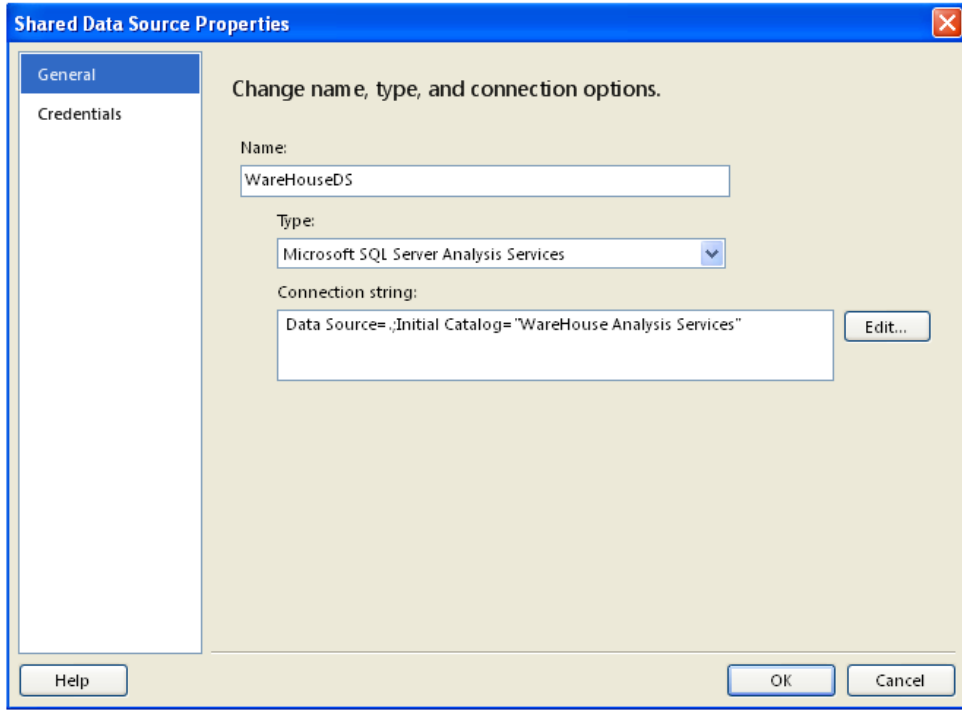
5.5 Karar Destek Amaçlı Sorgulama ve Raporlama Aracı Gerçekleştirimi

Sorgulama ve raporlama aracının çalışabileceği örnek bir veri ambarı ve OLAP küpleri oluşturulduktan sonra raporlama aracı oluşturmak için Microsoft Reporting Services kullanılmıştır. Bu yazılım aracı ile karar destek amacıyla karar alıcı konumundaki kişiler için, OLAP küpleri anlaşılır arayüzler içerisinde sunulması sağlanmıştır. Sorgulama dili olarak MDX sorgu cümlecikleri kullanılmıştır. OLAP küplerinin sorgulanabilmesi için uygun MDX sorgularının üretilmesi sağlanmıştır.



Şekil 5.17 Reporting Services Ekranı

Şekil 5.17' de Solution Explorer bölümü gösterilmektedir. Bir rapor projesi yaratılırken varsayılan olarak gelen Shared Data Sources altında raporun kullandığı veri kaynakları tanımlanır. Bu veri kaynakları SQL Server, Oracle, MS Access olabildiği gibi XML dosyası da olabilir. Bu tez uygulamasında veri kaynağı olarak Analysis Services seçilmiştir.

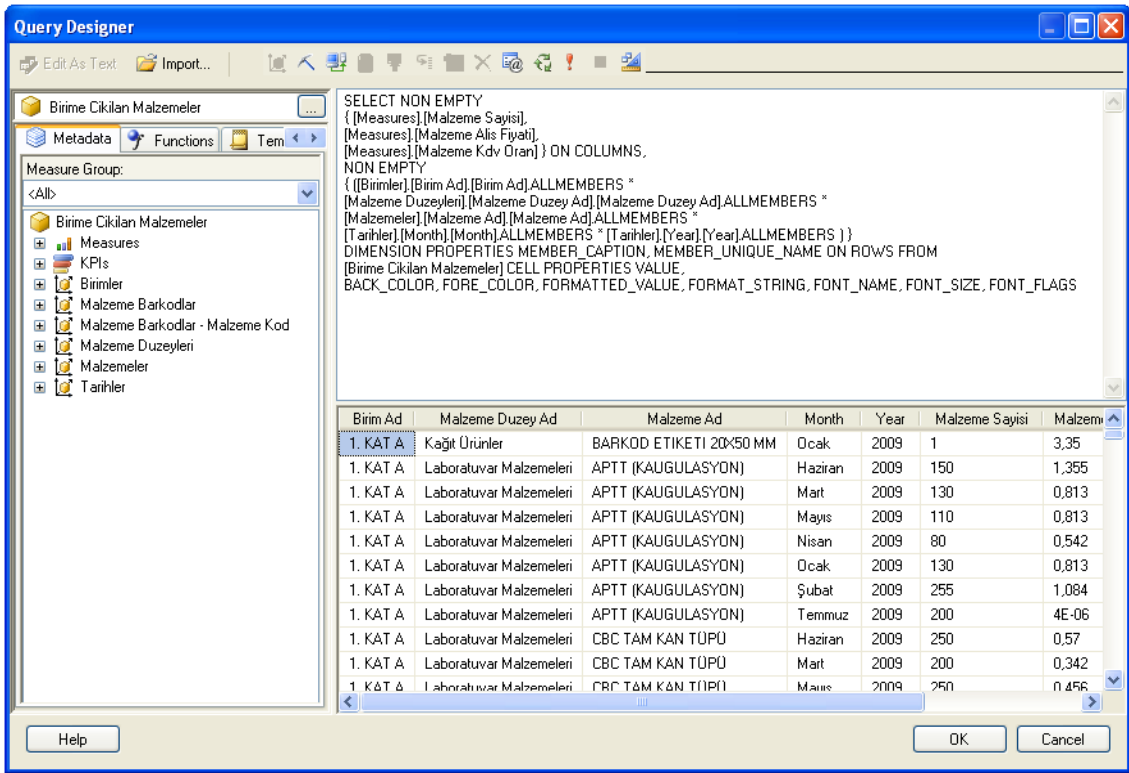


Şekil 5.18 Reporting Services Ekranından Analysis Services'e Bağlanma

Şekil 5.18'de görüldüğü gibi açılan pencerede bir datasource name, type ve connection string girilmesi gerekmektedir. Datasource name olarak "WareHouseDS" verilmiştir. Type kısmında ise Microsoft SQL Server Analysis Services'i seçilerek oluşturulan MDX sorgularının kullanılması sağlanmıştır. Connection string olarak "Data Source=.;Initial Catalog='WareHouse Analysis Services'" ifadesi girilerek Analysis Services'de oluşturulan OLAP veritabanındaki boyut ve küplere bağlanılmış olur. Veri ambarı ve OLAP sunucularına erişim için kullanılan bağlantı cümleciklerinde yer alan parametreler değiştirilerek fiziksel olarak farklı bilgisayarlarda bulunan sunuculara ve farklı kataloglara erişim sağlanabilmektedir.

Şekil 5.19 de görülen Query Designer ekranından raporlama işlemi için kullanılacak olan OLAP küp seçimi yapılmaktadır.

Şekil 5.18' de küp seçim işlemi yapıldıktan sonra aynı ekrandan MDX sorgulamaları yazılarak istenilen rapor alanlarına erişilebilir. Şekil 5.20' da örnek olarak verilen "Birime Çıkılan Malzemeler" küpü kullanılarak yazılmış olan MDX sorgusu görülmektedir.

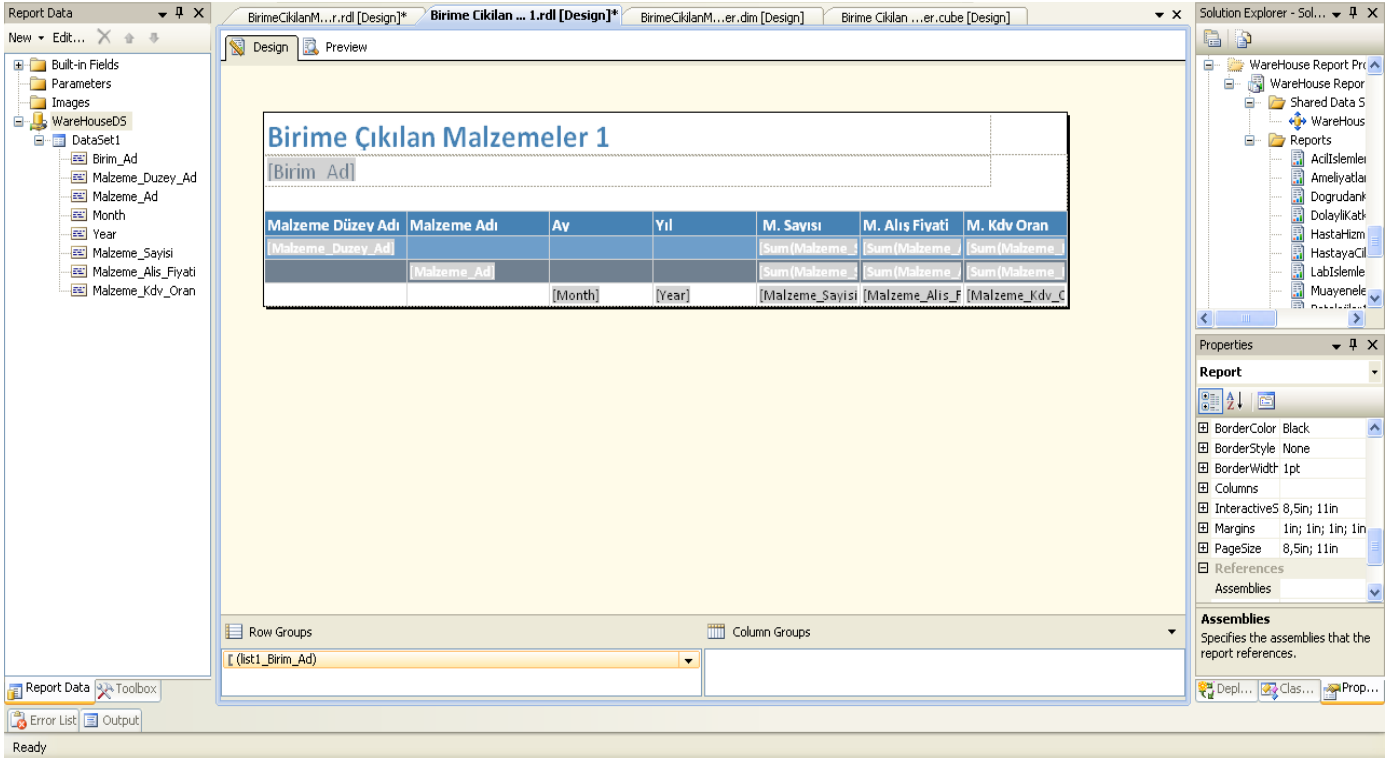


Şekil 5.19 Raporlama için OLAP Küp Seçimi

```
SELECT NON EMPTY
{ [Measures].[Malzeme Sayisi],
[Measures].[Malzeme Alis Fiyati],
[Measures].[Malzeme Kdv Oran] } ON COLUMNS,
NON EMPTY
{ ([Birimler].[Birim Ad].[Birim Ad].ALLMEMBERS *
[Malzeme Duzeyleri].[Malzeme Duzey Ad].[Malzeme Duzey Ad].ALLMEMBERS *
[Malzemeler].[Malzeme Ad].[Malzeme Ad].ALLMEMBERS *
[Tarihler].[Month].[Month].ALLMEMBERS * [Tarihler].[Year].[Year].ALLMEMBERS ) }
DIMENSION PROPERTIES MEMBER_CAPTION, MEMBER_UNIQUE_NAME ON ROWS
FROM
[Birime Cikilan Malzemeler] CELL PROPERTIES VALUE,
BACK_COLOR, FORE_COLOR, FORMATTED_VALUE, FORMAT_STRING, FONT_NAME,
FONT_SIZE, FONT_FLAGS
```

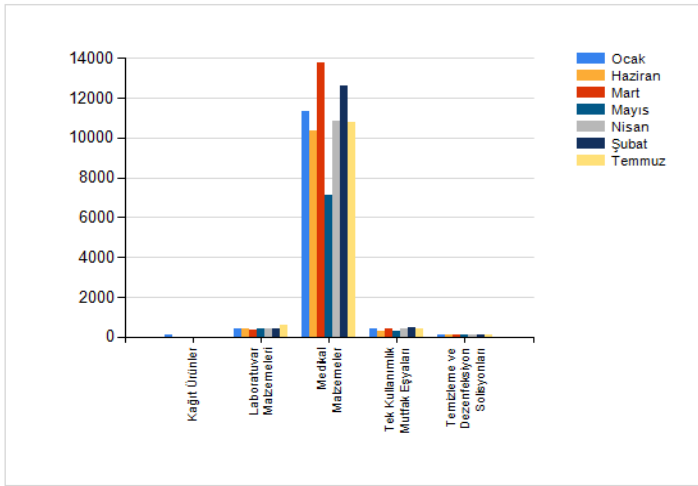
Şekil 5.20 “Birime Çıkılan Malzemeler” kübü kullanılarak oluşturulan MDX sorgulaması

Şekil 5.21’de gösterilen reports klasörü RDL (Report Definition Language) dosyalarının konulacağı alandır. Bir RDL dosyası oluşturulduğu zaman tasarım ekranının üst kısmında Data, Layout ve Preview sekmeleri bulunur. Data sekmesinde raporun kullandığı veri kümeleri (DataSet) gösterilir. Bir rapor birden fazla veri kümesinden oluşabilir. Bu veri kümeleri seçilerek daha önceden oluşturulan MDX sorguları bu ekranda düzenlenebilir.



Şekil 5.21 Birime Çıkılan Malzemeler Rapor Örneği

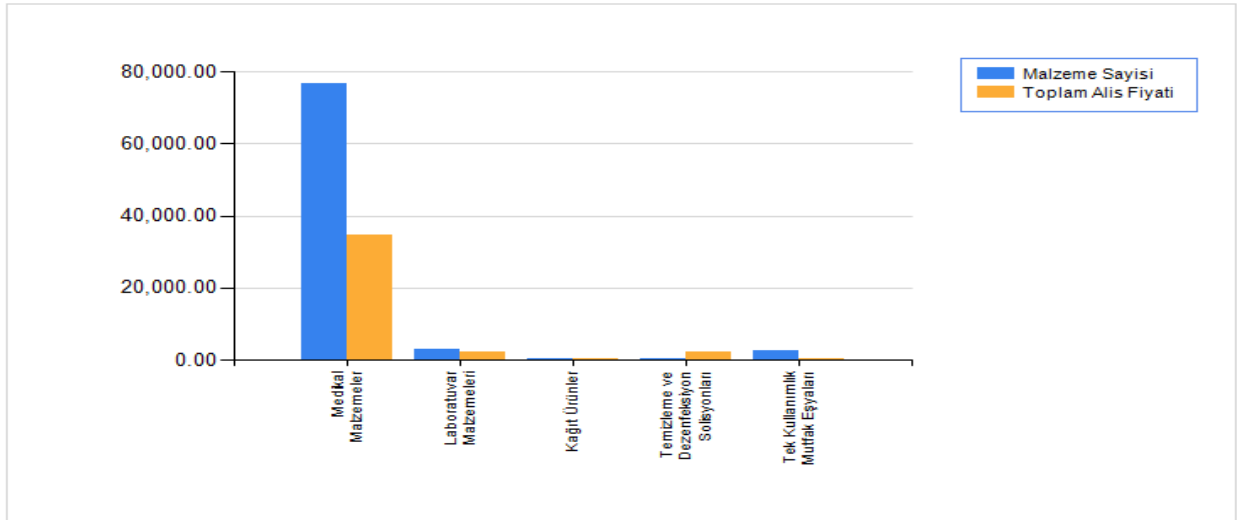
Bütün bu işlemlerden sonra hazırlanan raporlar deploy işlemi ile rapor sunucusuna (Report Server) yüklenir. Bu işlemden sonra sunucu üzerinde kurulu olan raporlama sunucusuna bağlanıp sunucuya ait veritabanı üzerinde ilgili kayıt işlemi gerçekleşir. Reporting Services “ReportServer” ve “ReportserverTempDB” olmak üzere iki veritabanına yükleme işlemini gerçekleştirir. İlk veritabanı RDL dosyalarıyla ilgili bilgileri içerir. İkinci veritabanı ise çok kullanıcı sistemlerde performans amacıyla ön belleğe alınmış raporların tutulduğu yerdir. Microsoft’un bu raporlama sunucusu bir web servisi gibi çalıştığı için dağıtık uygulamalara kolayca hizmet verir. Bu raporlara web ortamından erişim ve Microsoft Office 2007 ürünleriyle birlikte çalışması imkânı sağlanmıştır.



1. KAT A

Malzeme Düzey Adı	Malzeme Adı	Ay	Yıl	M. Alış Fiyatı	M. Kdv Oranı	M. Sayısı	M. Toplam Alış Fiyatı
☐ Kağıt Ürünler				3.35	18	1	3.953
☐ Laboratuvar Malzemeleri				63.832004	560	2930	2332.413056
☐ Medikal Malzemeler				1009.47013	8250	76762	34492.638004
☐ Tek Kullanımlık Mutfak Eşyaları				0.2574	468	2633	165.914496
☐ Temizleme ve Dezenfeksiyon Solisyonları				383.868	954	115	2051.80024

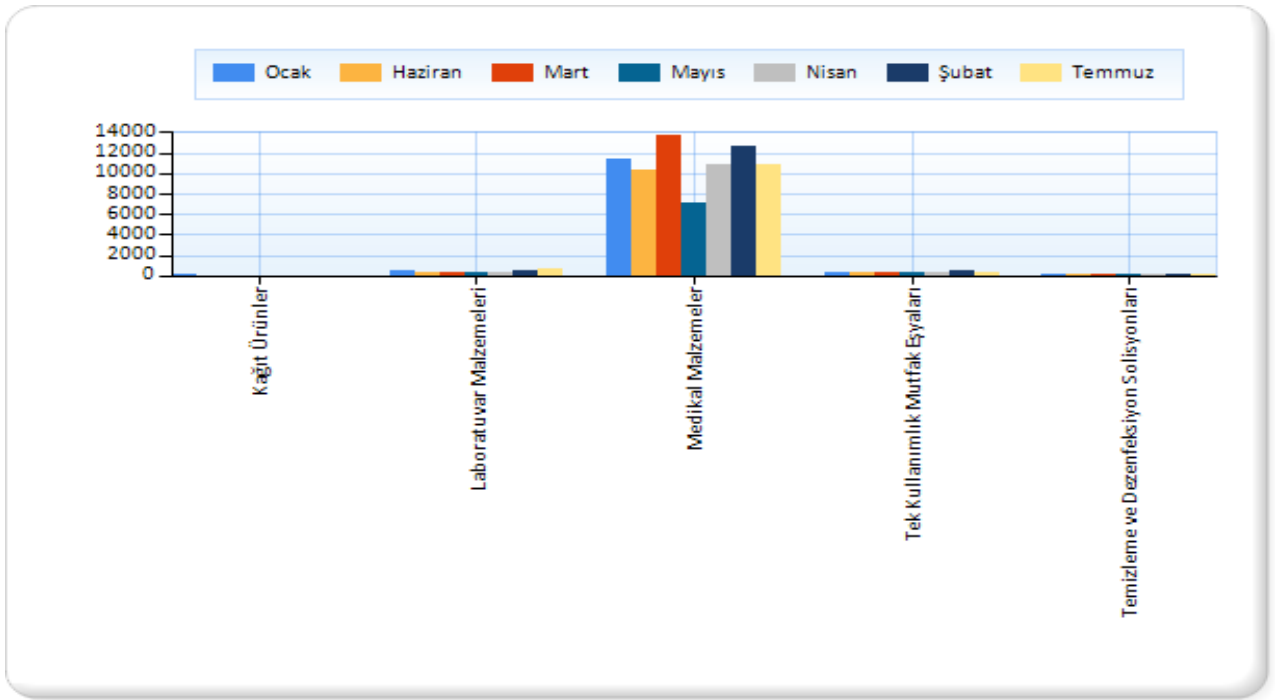
Şekil 5.22 Birime Çıkılan Malzemeler Aylara Göre Satış Sayıları



1. KAT A

	2009			
	Malzeme Sayısı	Malzeme Alış Fiyatı	Malzeme Kdv Oran	Toplam Alış Fiyatı
☐ Kağıt Ürünler	1	3.35	18	3.953
☐ Laboratuvar Malzemeleri	2930	63.832004	560	2332.413056
☐ Medikal Malzemeler	76762	1009.47013	8250	34492.638004
☐ Tek Kullanımlık Mutfak Eşyaları	2633	0.2574	468	165.914496
☐ Temizleme ve Dezenfeksiyon Solisyonları	115	383.868	954	2051.80024

Şekil 5.23 Birime Çıkılan Malzemeler Düzey Adlarına Göre Satış Sayıları



Şekil 5.24 Birime Çıkılan Malzemeler Düzey Adlarına Göre Toplam Satış Sayıları

Şekil 5.22-5.24'te program üzerinden alınabilecek raporlamalar görülmektedir. Raporlarda grafik olarak aylık, yıllık veriler düzey kodlarına ve bölümlere göre seçilebilmekte ve sıralanabilmektedir.

5.5.2 MDX Sorgulama

Sorgulama amaçlı en genel tanım aşağıdaki gibidir,

[WITH <Formül İfadesi>]

SELECT

[{ ... : ... : ... } on columns,

[{ ... : ... : ... } on rows]]

FROM [Küp İfadesi] WHERE [Koşul İfadesi]

“SELECT, FROM, WHERE” ifadeleri SQL sorgulama dilinde kullanılan yapıya benzese bile MDX ile daha karışık işlemler yapılabilir.

“SELECT, FROM, ON COLUMNS, ON ROWS” terimleri anahtar kelimelerdir. Köşeli parantez “[]” içerisinde tanımlanan terimler seçime bağlıdır.

“SELECT” ifadesi gösterilmek istenen verileri getirme işlemi için kullanılır. Ancak burada getirilecek olan veriler eksenler üzerinde bulunan hiyerarşik bir yapıda bulunan boyutlar üzerinden getirilmesi için de kullanılabilir. Her ne kadar SQL ile benzerliği olsa da SQL de kullanılan bu ifade ilişkisel veritabanlarından bize tablo geriye döndürür buna karşın MDX de kullanılan bu ifade OLAP veritabanından bize küp sonucu döndürecektir. Örneğin sütunlarda ilk önce kişilerin ait oldukları milliyetleri daha sonra da milliyetleri içerisinde cinsiyetlerini getirirken üst sırada yıllık gelirleri getirilebilir.

MDX “FROM” ifadesi tablo isimlerini tutmak için kullanılan SQL ifadesine benzer şekilde üzerinde işlem yapılacak olan kaynak küp ismini tutmak için kullanılır.

“WHERE” ifadesi seçme (slicer) işlemleri için kullanılır. SQL ifadesi olarak kullanıldığında arama işlemi olarak kullanıldığı unutulmamalıdır. Seçme işlemlerinde seçimi yapılan ekseninde azalma olmayacaktır, ancak seçilen eksene göre verilerde değişme olur.

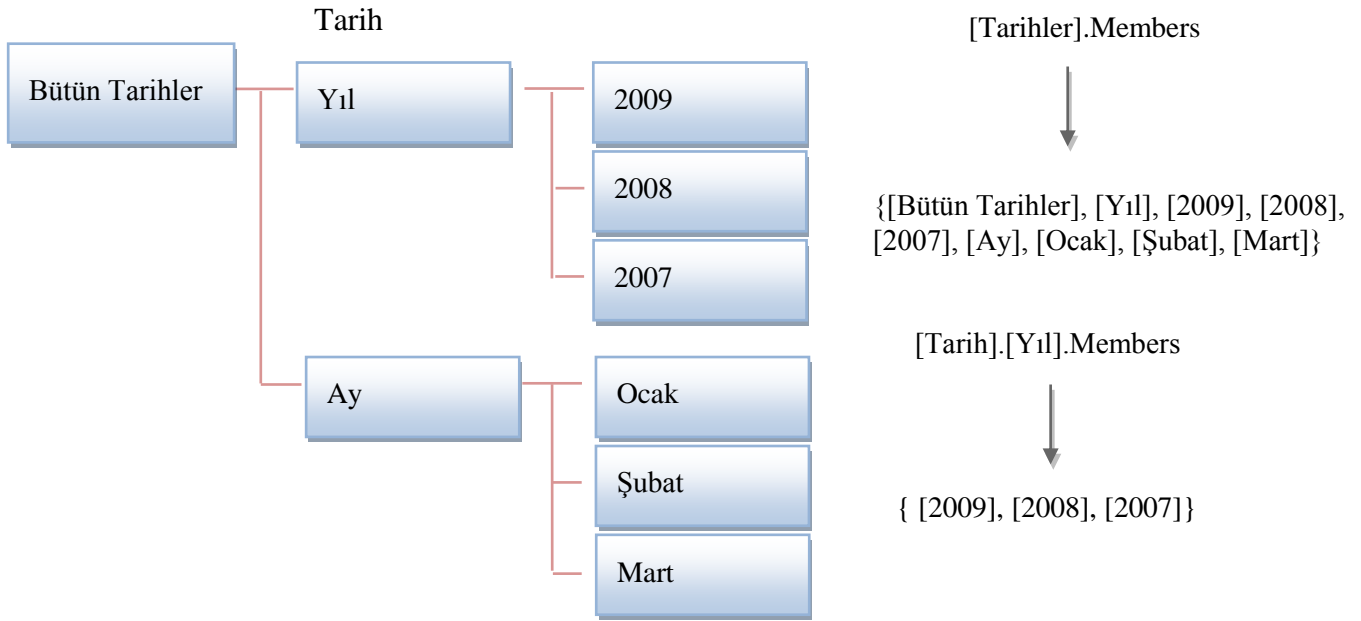
“ON COLUMNS” terimleri öncesinde x eksenini üzerinde, “ON ROWS” terimleri öncesinde de y eksenini üzerinde yer almasını istenilen boyut değişkenleri verilir. Köşeli parantezler içerisine sırasıyla boyutun adı, sıradüzeni, seviyesi ve özellikleri yazılır. Eğer boyut tek bir sıradüzenden oluşuyorsa boyut adından sonra doğrudan seviyeye geçilir. İlgili boyuta ait tüm seviyeler görüntülenmek isteniyorsa sadece boyut adı verilir.

Şekil 5.25’te belirtile “Tarihler” boyutunu ele alalım. “Ay, Yıl” şeklinde seviyelerimiz bulunsun.

[Tarihler].Members ifadesi, tarih boyutundaki tüm öğelerin elde edilmesini sağlar.

[Tarihler].[2009].Children ifadesi Tarih boyutunun 2009 yılına göre Ay seviyesine ait verilen seviyelere ulaşımı sağlayacaktır.

[Tarihler].[2009].[Ocak] ifadesi, Tarih boyutunun 2009 yılına ait Ocak bilgilerine erişimini sağlar.



Şekil 5.25 “.Members” Operatörü ile seçilmiş bütün üyeler

“SELECT

{[HastayaCikilanMalzeme Tarihler].[2009].Children } ON COLUMNS,

{ [HastayaCikilanMalzeme Hastalar].Members } ON ROWS

FROM [Hastaya Cikilan Malzemeler]” ifadesi ile “Hastaya Cikilan Malzemeler” küpü kapsamında, X eksenini üzerinde tarih boyutunun yıl seviyesine göre 2009 yılına ait değerler yer alacaktır. Y eksenini ise “HastayaCikilanMalzeme Hastalar” boyutunun tüm elemanları yer alacaktır ve x ve y eksenlerinin kesişim noktasını veren hücreler üzerinde iki boyut değerine karşılık gelen ölçüm değerleri görüntülenecektir.

MDX sorgulama işlemler kapsamında “WHERE” anahtar kelimesinin kullanımı ile elde edilecek sorgunun çeşitli kısıtlarla sınırlandırılması mümkündür.

Örneğin,

“SELECT

{[HastayaCikilanMalzeme Tarihler].[2009].Children } ON COLUMNS,

{[HastayaCikilanMalzeme Hastalar].Members } ON ROWS

FROM [Hastaya Cikilan Malzemeler] WHERE ([Hasta Tur Ad].[Ayaktan])“ ifadesiyle, “Hastaya Cikilan Malzemeler” küpü kapsamında sadece ayaktan hastalara çıkılmış malzeme ve ilaçların, x ekseninde 2009 yılına ait tarih boyutu ve y ekseninde “HastayaCikilanMalzeme Hastalar” boyutunun görüntülenmesini sağlayacaktır.

5.6 Veri Ambarı ve Veri Tabanı Karşılaştırılması

OLTP kapsamında ilişkisel veri tabanında tutulan veriler ve bu veriler üzerinde çalışan raporlar ile OLAP kapsamında veri ambarında ve çok boyutlu küplerde tutulan veriler ve bu veriler üzerinde çalışan raporların karşılaştırmasında aşağıdaki veriler elde edilmiştir.

Şekil 5.26’te veri ambarı ve veri tabanı arasındaki veriler karşılaştırılmıştır. Toplam tablo sayısı ilişkisel veri tabanında toplam 408 adettir. Bu sayı yapılan yeni uygulamalarla artması kaçınılmazdır. Bu tabloların kapladığı alan 60 GigaByte’tan fazladır. Bu tablolardan veri ambarında kullanılan raporlamalar çekilmek istendiğinde en iyi ihtimalle 1 aylık çekilen sorgu gelme süresi 45 dakikayı bulmaktadır. Bu sorgulama işlemi sırasında aktif kullanıcı sayısı 300 kişiden fazladır. Eğer bu sorgulamalar aktif kullanıcıların sistemde olmadığı zamanda çalıştırılırsa aynı sorgular 25 dakikada gelmektedir. Bu sorgulamaların sadece 1 aylık olduğu ve sadece ham verileri içerdiği unutulmamalıdır.

Örneğin “Uzman Doktor Doğrudan ve Dolaylı Performans Hesabı” için hazırlanan 3 aylık sorgulamalar ve bu sorgulara ait raporların düzenlenmesi 2 haftalık süreyi almaktadır.

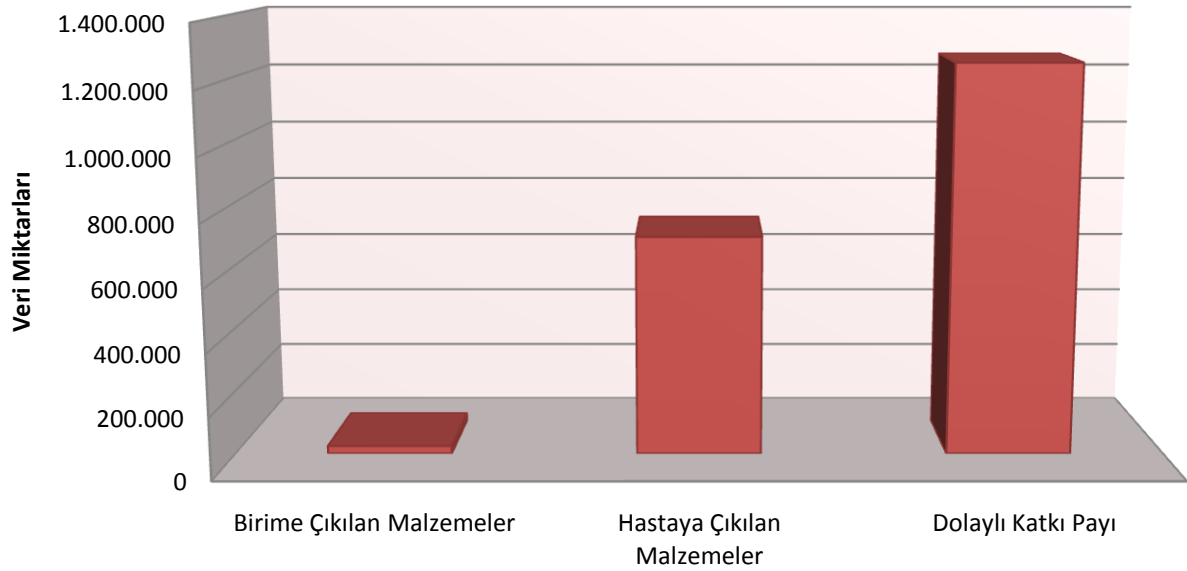
Veri ambarında raporlar için oluşturulan toplam tablo sayısı 46 adettir. Oluşturulan tabloların kapladığı alan 400 MegaByte’tan fazladır. Raporlar için oluşturulan sorgulamaların çalıştırılmasında veriler 10 dakikadan kısa sürede hazır hale gelmektedir. Bu raporlamalarda kullanıcıların veriler üzerinden istatistiksel sonuçlar çıkarılabilmesi için ayrıca bir is gücü harcanması gerekmemektedir.

Örneğin “Hastaya Çıkılan Malzemeler” için hazırlanan veritabanından 7 aylık sorgulama çekilmesi işlemi gelen işlenmiş ve raporlamaya hazır gelen yaklaşık 700 bin adet veri 1 dakikadan kısa sürede gelmektedir.

	Veri Tabanı	Veri Ambarı
Toplam tablo sayısı	408	46
Toplam Veri Sığası	60 GB	400 MB
Ortalama Kullanıcı Sayısı	300	5
Raporların Ortalama Çalışma Süresi	0-45dk (Aylık)	0-10dk (7 Aylık)

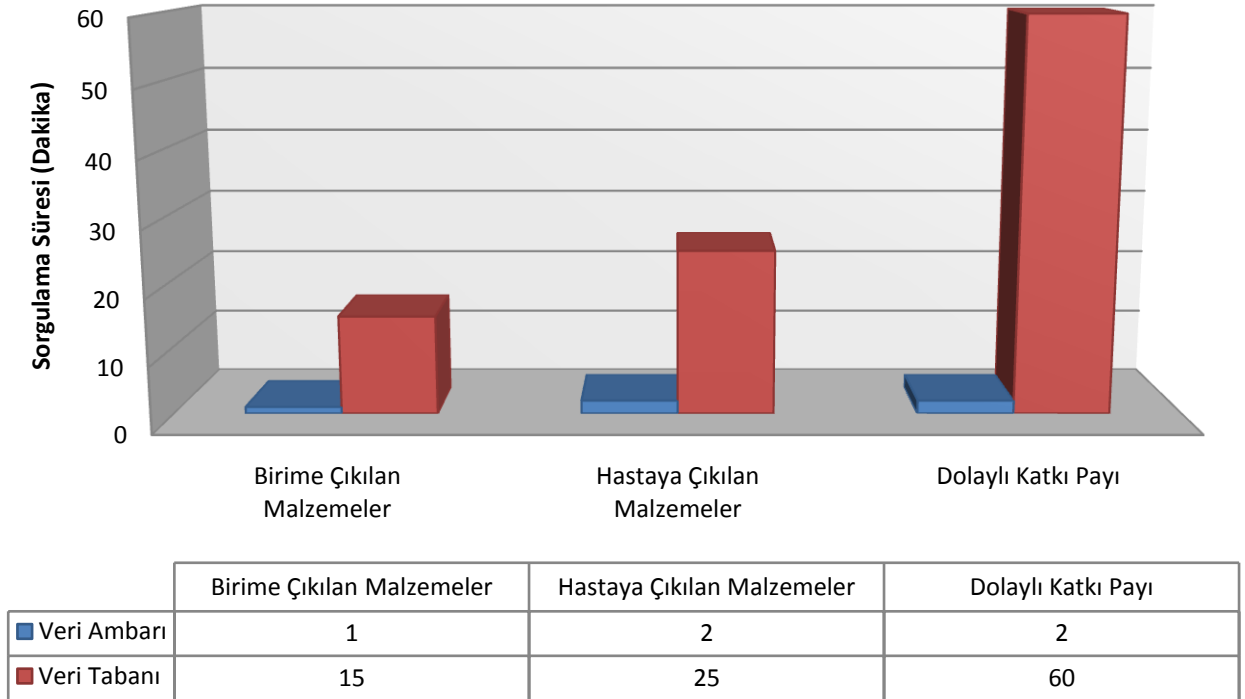
Şekil 5.26 Veri Ambarı ve Veri Tabanı Karşılaştırması

Şekil 5.27’de tez kapsamında kullanılan veri miktarları görülmektedir. Bu veriler HBYS ve MYS için kullanılan 2009 yılına ait 7 aylık veri miktarlarını göstermektedir. “Birime Çıkan Malzemeler” sorgusunda yaklaşık olarak 25.000 adet veri geldiği görülmektedir. Aynı şekilde “Hastaya Çıkan Malzemeler” için yaklaşık 710.000 ve “Dolaylı Katkı Payı” için 1.260.000 adet veri girişi gözlenmektedir.



	Birime Çıkan Malzemeler	Hastaya Çıkan Malzemeler	Dolaylı Katkı Payı
■ Veri Miktarı (Yaklaşık)	25.000	710.000	1.260.000

Şekil 5.27 Sorgulamada Gelen Veri Miktarları



Şekil 5.28 Veri Ambarı ve Veri Tabanı Sorgulama Süreleri

Şekil 5.28’de tez kapsamında kullanılan verilere erişim süreleri görülmektedir. “Birime Çıkılan Malzemeler” raporlaması için kullanılacak yaklaşık 25.000 adet veriye veri ambarından Analysis Services içerisinde MDX sorgulaması kullanılarak 1 dakikadan az sürede erişilebilmektedir. Aynı raporlama için kullanılacak verilere veri tabanından SQL sorgulama dili ile 15 dakikada erişilmektedir. Aynı şekilde “Hastaya Çıkılan Malzemeler” raporlaması için veri ambarı kullanılarak 2 dakika, veri tabanı kullanılarak 25 dakika ve “Dolaylı Katkı Payı” için ise veri ambarı için 2 dakika, veri tabanı için yaklaşık olarak 60 dakika beklemek gerekmektedir.

Bu sorgulamalar yaklaşık olarak 320 adet kullanıcının sistemi kullandığı zamanlarda tekrarlanmıştır. Veriler ortalama olarak alınmıştır.

6. SONUÇLAR VE DEĞERLENDİRME

Bu proje kapsamında incelenen veri ambarı ve OLAP teknolojileri yüksek hacimli veri üzerinde sorgulama ve çok boyutlu analiz ihtiyacını karşılamak için ortaya konan çözümlerdir. Veri ambarında tutulan ilişkisel tablolardan çok boyutlu küplere geçiş sağlanarak OLAP teknolojileri ile verinin çok daha kullanışlı olması, verinin anlamlılığının artırılması ve veriye hızlı erişim sağlanmıştır.

Proje kapsamında geliştirilen uygulama sayesinde kullanıcılar istedikleri veriyi kendi istekleri doğrultuda hazırladıkları bir biçim içinde görebilir hale gelmişlerdir. Ayrıca normal raporlama araçları ile yapılması zor ve zaman alan raporları kullanıcılar kendileri yapabilir hale gelmişlerdir. Bu açıdan kullanıcı memnuniyeti sağlanmıştır. Kullanıcıların veriye erişimi operasyonel sistem üzerine herhangi ek yük getirmemektedir. Bu açıdan sistem yöneticilerinin memnuniyeti sağlanmıştır. Bu proje sayesinde ek bir yatırım yapmadan bu teknolojinin getireceği faydaların gözlenebileceği düşünülmektedir. Şu anda OLAP istemcisi olarak kullanılan Microsoft Reporting Services ürünü, raporlara web servisi üzerinden erişim imkânı sağlar. Gerekli ayarların yapılmasıyla kullanıcıların istedikleri ortamdan raporlara erişimi sağlanmış olur.

Bir veri ambarından beklenen, sorguların çok hızlı çalışması ve sorguların kolayca tanımlanması gibi özelliklere sahip olmasıdır. Bu bakımdan HBYS için geliştirilen veri ambarında ihtiyaç duyulan bilgiye kolaylıkla ulaşılmakta ve sorgulamaların çok hızlı çalıştığı görülmektedir.

Çalışmanın bir sonraki safhası ise bulanık mantık, genetik algoritma, sinir ağı gibi uygulamalar ile geçmiş verilerden yola çıkılarak ileriye yönelik verilerin tahmin edilmesi amacıyla veri madenciliği yapılmasıdır.

KAYNAKLAR

- Ahmad, I., Azhar, S., Lukauskis, P., (2004), Development of A Decision Support System Using Data Warehousing to Assist Builders/Developers in Site Selection, Decision Support Systems.
- Aggarwal,S. ve van der Valk,H (2008), Data Warehouse Performance: Unisys and Microsoft Achieve Record-setting Benchmark, Netherlands.
- Aköz, E. (2007), Otomotiv Sektöründe Veri Ambarı ve Bir Uygulama, İstanbul.
- Bayoğlu, L. (2000). Management Information System and Data Warehousing, İzmir.
- Birant, D. (2006), Modeling and Analyzing Marine Data Using Data Mining Techniques, İzmir.
- Çetinyokuş, T. ve Gökçen, H. (2007), Bütünleşik Veri Kupu Sistemi (BVKS): Satış Kupu Uygulaması, İstanbul.
- Döşlü, A. (2008), Veri Madenciliğinde Market Sepet Analizi ve Birlikte Kuramlarının Belirlenmesi, İstanbul.
- Güleç, F. G. (2007) Kurumsal Verilerin Yapay Zekâ Modelleri ile İşlenmesi İçin Modelleme Aracı Alt Yapı Tasarım ve Gerçekleştirimi, Ankara.
- Güratan, I. (2005), The Design and Development of a Data Warehouse Using Sales Database and Requirements of a Retail Group, İzmir.
- Gray ,J., Chaudhuri, S., Bosworth, A., Layman A., Reichart, D., Venkatrao, M.,(1997), Data Mining and Knowledge Discovery.
- Han, J. and KAMBER, M. (2001), Data Mining: Concepts and Techniques, Morgan Kaufmann Publishers, USA.
- Hoyur,G. A. (2002) Simulation Model for Distribution Warehouses, İstanbul.
- Kimball, R. (2002) The Datawarehouse Toolkit Olap Solutions ISBN 0471200247.
- Kuğu, E (2005), Oracle OLAP (On-Line Analytical Processing) Teknolojisi Kullanılarak Bir Filodaki Faal Uçak Sayısının Uçuş Saatlerine Etkilerinin Gözlemlenmesi, İstanbul.
- Seidman, C.(2001), Data Mining with Microsoft SQL Server 2000, Microsoft Press
- Tantuğ, A. C. (2002), Veri Madenciliği ve Demetleme, Yüksek Lisans Tezi, İstanbul Teknik Üniversitesi Fen Bilimleri Enstitüsü, İstanbul.
- Turleyet, P. (2007) Microsoft SQL Server 2005 Integration Services Step by Step ISBN:9780735624054

ÖZGEÇMİŞ

Devrim İŞLİ, 1980 yılında Denizli’de doğdu. 2003 yılında İzmir Yüksek Teknoloji Enstitüsü, Bilgisayar Mühendisliği bölümünden mezun olmuştur. 2006 yılından beri Pamukkale Üniversite Hastanesi Bilgi İşlem Bölümünde Uzman olarak sistem analizi ve yazılım geliştirme alanında çalışmaktadır. HBYS ve MYS projelerinde görev almıştır. Aynı zamanda Pamukkale Üniversitesi Sürekli Eğitim Merkezi’nde programlama ve veritabanı eğitimleri vermektedir. Askerlik görevini 2004 yılında İzmir’de tamamlamıştır.