

T.C.  
BEYKENT ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ  
MATEMATİK BİLGİSAYAR ANA BİLİM DALI  
BİLGİSAYAR AĞLARI VE İNTERNET TEKNOLOJİLERİ  
BİLİM DALI

**OTOMOTİV SEKTÖRÜNDE VERİ AMBARI  
VE BİR UYGULAMA  
(YÜKSEK LİSANS TEZİ)**

Tezi Hazırlayan:

**Erhan AKÖZ**

Öğrenci No:

050861010

Danışman:

Dr. Yalçın ÖZKAN

İSTANBUL, 2007

## YEMİN METNİ

Sunduđum Yüksek Lisans Tezimi, Akademik Etik İlkelerine bađlı kalarak, hi kimseden akademik ilkelere aykırı bir yardım almaksızın bizzat kendimin hazırladıđına and içerim. *19.1.2027*

Aday: Erhan AKÖZ



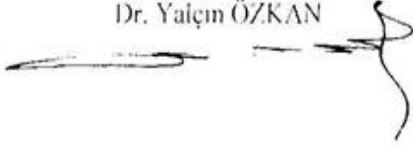
T.C.  
BEYKENT ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ MÜDÜRLÜĞÜ  
TEZLİ YÜKSEK LİSANS TEZ SINAV TUTANAĞI

..19.09.2007

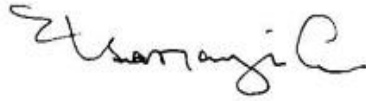
Enstitümüz Matematik Bilgisayar Anabilim Dalı Bilgisayar Ağları ve İnternet Teknolojileri Bilim Dalı yüksek lisans öğrencilerinden **050861010** numaralı **Erhan AKÖZ**'ün "Beykent Üniversitesi Lisansüstü Eğitim - Öğretim ve Sınav Yönetmeliği'nin ilgili maddesine göre hazırlayarak Enstitümüze teslim ettiği "**OTOMOTİV SEKTÖRÜNDE VERİ AMBARI VE BİR UYGULAMA**" başlıklı tezi/projesi, Yönetim Kurulumuzun 06.08.2007 tarih ve 2007/10 sayılı toplantısında seçilen ve Fakülte binasında toplanan biz jüri üyeleri huzurunda, ilgili yönetmeliğin (c) bendi gereğince (.....) dakika süre ile aday tarafından savunulmuş ve sonuçta adayın tezi/projesi hakkında **Oyçokluğu/Oybirliği** ile **Kabul/Red veya Düzeltme** kararı verilmiştir.

İşbu tutanak, 4 nüsha olarak hazırlanmış ve Enstitü Müdürlüğü'ne sunulmak üzere tarafımızdan düzenlenmiştir.

DANIŞMAN  
Dr. Yalçın ÖZKAN



ÜYE  
Prof.Dr. Esat HAMZAOĞLU



ÜYE  
Yrd.Doç.Dr.Zuhal TANRIKULU



# OTOMOTİV SEKTÖRÜNDE VERİ AMBARI VE BİR UYGULAMA

**Tezi Hazırlayan: Erhan AKÖZ**

## Özet

Otomotiv Sektöründe Veri Ambarı ve Bir Uygulama” isimli çalışmamızda, Veri Ambarı bilgi teknolojisi tanıtılmaya çalışılmıştır. Çalışma konumuzun esasını oluşturan Veri Ambarı bilgi teknolojisi, Microsoft SQL Server 2000’in işlevselliğinden yararlanılarak Doğu Otomotiv Servis Ticaret A.Ş. Satış Sonrası Hizmetleri operasyonel verilerine uygulanarak bir Veri Ambarı oluşturulmuştur. Çalışmamızda; Doğu Otomotiv Servis Ticaret A.Ş. Satış Sonrası Hizmetlerde mevcut operasyonlarında kullanılan veri akışı ve yönetimi incelenmiş olup, veriler toplanıp, bu bilgilerin Veri Ambarına dönüştürülmesi ve üzerinde çeşitli analizler yapılmasına olanak sağlayan raporlar tasarlanmıştır.

**Anahtar Kelimeler:** Otomotiv Sektörü, Satış Sonrası Hizmetler, Veri Ambarı

# **DATA WAREHOUSE IN THE AUTOMOTIVE SECTOR AND AN APPLICATION**

**Presented by: Erhan AKÖZ**

## **Abstract**

The Data Warehouse information technique has been utilized in this publication and is specifically referred to as “Data Warehouse and Application in the Automotive Sector”. Towards this aim, the operational datas have been obtained from the After Sales Services of Doğuř Otomotiv Company and The Data Warehouse developed by Microsoft SQL Server 2000. In our study, data flow and management was analyzed and collected from the current operations of After Sales Services of the Doğuř Otomotiv, based on and using that data, a Data Warehouse has been produced. Some reports are designed to give management the ability to analyze the capability of the Data Warehouse.

**Key Words: Automotive Sector, After Sales Services, Data Warehouse**

## İÇİNDEKİLER

Yemin Metni	
Juri Sayfası	
Türkçe Özet ve Anahtar Kelimeler	
İngilizce Özet ve Anahtar Kelimeler (Abstract)	
Şekiller Listesi	VI
Kısaltmalar	VIII
<b>Giriş</b>	<b>1</b>

### I. BÖLÜM

#### VERİ AMBARI

1.1. Veri Ambarı Nedir	2
1.2. Veri Ambarı Tarihi	5
1.3. Veri Ambarı Oluşturmak için Nedenler	6
1.4. Veri Ambarı Hedefi	7
1.5. Veri Ambarı Kullanıcıları	8
1.6. Veri Ambarı Özellikleri ve Üstünlükleri	8
1.7. Veri Ambarı Kavramsal Tasarımı	9
1.7.1. Yıldız Şema	10
1.7.2. Kar tanesi Şeması	12
1.7.3. Takım Yıldızı Olgusu	13
1.8. Ambar Mimarisi	14
1.8.1. Veri Ambarı Mimarisi (Temel)	14
1.8.2. Veri Ambarı Mimarisi (Aşamalı Alan)	15
1.8.3. Veri Ambarı Mimarisi(Aşamalı Alan ve Data Mart)	16
1.9. Veri Ambarı Sunucu İstemci Araçları ve Yardımcı Uygulamalar	17
1.10. Ayrı Veri Ambarı Oluşturma Nedenleri	18
1.11. Data Mart	19
1.12. Data Mart Oluşturmak İçin Nedenler	19
1.13. Data Mart ve Veri Ambarı Farkları	19

## **II. BÖLÜM**

### **OTOMOTİV SEKTÖRÜ**

<b>2.1. Dünya Otomotiv Sanayii</b>	<b>21</b>
<b>2.2. Türkiye Otomotiv Sanayii</b>	<b>22</b>
<b>2.3. Dođuş Otomotiv Servis Ticaret A.Ş Tarihçe</b>	<b>24</b>
<b>2.4. Servis İş Modeli.</b>	<b>26</b>
2.4.1. Randevu İşlemi	27
2.4.2. İş Emri Açılması	28
2.4.3. İş Emri Tamamlanması	29

## **III. BÖLÜM**

### **VERİ AMBARI TASARIMI**

<b>3.1. Veri Ambarı Kavramsal Model Tasarımı</b>	<b>30</b>
3.1.1. Boyut Tabloları	32
3.1.1.1. Marka Kodu	32
3.1.1.2. Kampanya	32
3.1.1.3. Müşteri Araç İlişkisi	33
3.1.1.4. Müşteri	34
3.1.1.5. Eğitim	35
3.1.1.6. Kurum	36
3.1.1.7. Personel	36
3.1.1.8. Statü	37
3.1.1.9. Statü Tipi	38
3.1.1.10. Zaman	38
3.1.1.11. Araç	39
3.1.1.12. Garanti Tipi	40
3.1.2. Olgu Tabloları	41
3.1.2.1. Randevu Tablosu	41
3.1.2.2. İş Emri Tablosu	42
3.1.2.3. İş Emri Detay Tablosu	44
<b>3.2. Veri Akışı</b>	<b>44</b>
3.2.1. ODBC Kaynağı Tanımlanması	45
3.2.2. Veri Kaynağı Yapılandırılması	46

3.2.3. DataIntegrator da Datastore tanımı oluşturmak	47
3.2.4. Seçip Çıkarma İşlemi	49
3.2.5. DüzDosya Örneği	51
3.2.6. Dönüşüm İşlemi	51
3.2.7. Yükleme İşlemi	52
3.2.8. Boyut Tabloların Doldurulması	52
3.2.9. Universe oluşturmak	53
3.2.9.1.Universe bağlantısı oluşturmak	53
<b>3.3. Raporlama</b>	<b>55</b>
3.3.1. Business Objects'te Örnek Rapor Oluşturmak	56
3.3.2. Veri Ambarı Raporları	59
<b>Sonuç</b>	<b>62</b>
<b>Kaynaklar</b>	<b>63</b>
<b>Özgeçmiş</b>	<b>64</b>



## ŞEKİLLER LİSTESİ

<u>Şekil No</u>	<u>Sayfa</u>
Şekil 1: Yıldız Şema	11
Şekil 2: Kar tanesi Şeması	12
Şekil 3: Takımyıldızı Olgusu	13
Şekil 4: Veri Ambarı Mimarisi	14
Şekil 5: Aşamalı Alanlı Veri Ambarı Mimarisi	15
Şekil 6: Aşamalı Alan ve Data Martlı Veri Ambarı Mimarisi	16
Şekil 7: Servis Randevusu İş Akışı	27
Şekil 8: Servis Randevusu İş Emri Akışı	28
Şekil 9: İş Emri Tamamlama	29
Şekil 10: DOAS SSH Veri Ambarı Kavramsal Model Tasarımı	31
Şekil 11: Marka Kodu	32
Şekil 12: Kampanya	32
Şekil 13: Müşteri Araç İlişki Tablosu	33
Şekil 14: Müşteri Tablosu	34
Şekil 15: Eğitim	35
Şekil 16: Kurum	36
Şekil 17: Personel	37
Şekil 18: Statü	37
Şekil 19: Statü Tipi	38
Şekil 20: Zaman	38
Şekil 21: Araç Tablosu	40
Şekil 22: Garanti Tipi	40
Şekil 23: Randevu Tablosu	41
Şekil 24: İş Emri Tablosu	43
Şekil 25: İş Emri Detayı Tablosu	44
Şekil 26: Denetim Masasında ODBC Veri Kaynağı Simgesinin Seçimi	46

Şekil 27: Veri Kaynağı Yapılandırılması	46
Şekil 28: Veri Kaynağı Yapılandırılması	47
Şekil 29: Datastore tanımı oluşturmak	48
Şekil 30: Veri ambarı için Datastore tanımı oluşturmak	48
Şekil 31: Seçip Çıkarma İşlemi	49
Şekil 32: Seçip Çıkarma İşlemi	50
Şekil 33: DüzDosya Örneği	51
Şekil 34: Dönüşüm İşlemi	51
Şekil 35: Yükleme İşlemi	52
Şekil 36: Boyut Tablolarının Doldurulması	53
Şekil 37: Universe Bağlantısı Oluşturmak	54
Şekil 38: Universe Bağlantısı Oluşturmak	54
Şekil 39: Universe Oluşturma	55
Şekil 40: Business Objects XI Açılış Ekranı	56
Şekil 41: Business Objects XI Web Intelligence Dökümanı Oluşturmak	57
Şekil 42: Business Objects XI Web Intelligence Universe Seçim Ekranı	57
Şekil 43 Business Objects XI Rapor Tasarlama Ekranı	58
Şekil 44: Business Objects XI Örnek Rapor Ekranı	58
Şekil 45: Business Objects XI Rapor Tasarlama Ekranı Özellikleri Araçları	59
Şekil 46: Doğu Otomotiv Servis Ticaret A.Ş. 2000 Yılı SSH İş Emri Analiz Raporu	60
Şekil 47: Doğu Otomotiv Servis Ticaret A.Ş. 2000 Yılı SSH İş Emri Analiz Raporu Column Chart Versiyonu	60
Şekil 48: Doğu Otomotiv Servis Ticaret A.Ş. 2000 Yılı SSH İş Emri Analiz Raporu Pie Chart Versiyonu	61

## KISALTMALAR LİSTESİ

ADO	ActiveX Veri Nesnesi
a.g.k.	Adı Geçen Kitap
a.g.m.	Adı Geçen Makale
CRM	Müşteri İlişkileri Yönetimi
DBA	Veri Tabanı Yöneticisi
DBMS	Veri Tabanı Yönetim Sistemi
DM	Veri Madenciliği
DSS	Veri Ambarı Kullanıcısı
DTM	Dış Ticaret Müsteşarlığı
DW	Veri Ambarı
ERP	İşletme Kaynak Planlama
ETL	Seçip Çıkarma, Taşıma ve Dönüştürme
FTP	Dosya Aktarma Protokolü
IT	Bilişim Teknolojisi
KDS	Karar Destek Sistemi
MIS	Yönetim Bilgi Sistemi
MS SQL	Microsoft Yapısal Sorgu Dili
ODBC	Açık Veritabanı Bağlanırlığı
OLAP	Çevirim içi Analitik İşlem
OS	İşletim Sistemi
RDBMS	İlişkisel Veritabanı Yönetim Sistemi
SQL	Yapısal Sorgu Dili
SSH	Satış Sonrası Hizmetler



## GİRİŞ

İşletmelerde özellikle stratejik seviyedeki yönetim kademelerinde bilişim sistemlerinin kullanımı ile ilgili olarak sorunlar yaşanmaktadır. Üst düzeydeki yöneticilerin bilgi gereksinimini operasyonel düzeydeki bilişim sistemleri karşılayamamaktadır. Bu gereksinime paralel olarak “karar destek sistemleri” ortaya çıkmış ve günümüzde yaygın bir uygulama alanı bulmuştur. Diğer birçok sektörde olduğu gibi otomotiv sektöründe ve özellikle otomotiv pazarlama sektöründe, yoğun rekabet nedeniyle karar destek sistemlerinin kullanımı önem taşımaktadır.

Karar destek sistemleri ise geleneksel OLTP altyapısı ile çalışmamakta, özel birtakım yapılara gereksinim duyulmaktadır. Günümüzde karar destek sistemlerinin alt yapısı olarak Veri Ambarlarına başvurulmakta ve bu veri ambarına dayalı olarak gerekli karar çıktıları elde edilmektedir. Veri ambarı kurumun operasyonel sisteminden ayrı tasarlanmakta ve farklı bir veri modeli tercih edilmektedir. Uygulamada farklı sektörler için farklı veri ambarı modellerinin tasarlanması söz konusudur.

Bu çalışmada Otomotiv satış sonrası hizmetler sektörü için tasarladığımız bir veri modeli ortaya konulmaktadır. Söz konusu veri ambarı modeli inşa edilirken, uygulamanın amacı yanı sıra sistemin performansını maksimize edecek bir veri modeli oluşturulması amaçlanmıştır.

Çalışmanın birinci bölümünde veri ambarının ne olduğu üzerine kuramsal bilgiler verilmektedir. İkinci bölümde Türkiye’deki Otomotiv sektörü hakkında bilgi verdikten sonra, Doğu Otomotiv firmasının operasyonel sistemleri açıklanacak ve bu operasyonel sistemlerden yararlanılarak bir veri ambarı mimarisi tasarımı ortaya konulacaktır. Bu tasarımın şu şekilde bir sonuca ulaşması beklenmektedir: a) Sorgular çok hızlı çalışmalıdır. b) Anlık sorgular (ad-hoc query) kolayca tanımlanmalıdır.

# I. BÖLÜM

## VERİ AMBARI

### 1.1 Veri Ambarı Nedir?

Veri Ambarı'na ait literatür de pek çok tanım bulunmaktadır, bazılarını şöyle sıralayabiliriz:

“Veri Ambarı büyük hacimlerdeki tarihsel veriyi analiz etmek ve yönetim karar aşamasında kullanmak için uyarlanmış bir yaklaşımdır”<sup>1</sup>.

“Farklı yerlerde farklı kaynaklarda bulunan verilerin karar destek amaçlı kullanılmak üzere gerekli bilgilere dönüştürülmesini ve kullanıcılara dağıtılmasını sağlayan bir teknolojidir”<sup>2</sup>.

“Genel olarak, Veri Ambarı iş birimlerini veya çoklu ortak işlevleri destekleyen bir karar destek veritabanı olarak tanımlanmaktadır”<sup>3</sup>.

“Veri Ambarı, yönetimin karar verme sürecini desteklemede verinin konu-yönelimli, bütünlük, zaman değişkenli, değişken olmayan bir toplamıdır”<sup>4</sup>.

“Çeşitli bilgisayarlara dağıtılmış, çeşitli formatlarda olabilen bilgilere hızlı erişilebilmesini sağlayan ve karar verme sürecinde stratejik verilerin elde edilmesine olanak verecek şekilde düzenlenmiş merkezi veri yapılarıdır”<sup>5</sup>.

İlişkili bir veritabanı olan Veri Ambarı, hareket işleminden daha çok sorgu ve veri analizleri için tasarlanmıştır. Genellikle geçmiş verileri içermekte olan Veri Ambarı hareket verilerinden elde edilebileceği gibi diğer kaynaklardan da veri alabilir. Bu hareket iş yüklemesinden, analiz iş yüklemesini ayırarak, farklı kaynaklardan veri birleştirme organizasyonuna olanak sağlar.

---

<sup>1</sup> Yahiko K., Mukesh M., Wolfram Wob, Data Warehousing and Knowledge Discovery, 2004, Spain s.24.

<sup>2</sup> Çarkıt N., “Yönetim Süreçlerinde Bilgisayara Dayalı Karar Destek Sistemlerinin İrdelenmesi”, G.Ü. Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, Ankara, 1997, s.43.

<sup>3</sup> Craig R., Vivona J., Berkovitch D., “Microsoft Data Warehousing.” Building Distributed Decision Support Systems, Mart, 1999. s.47.

<sup>4</sup> Inmon W.H., What is a Data Warehouse? W.H. Inmon, Prism, 1995, s.4.

<sup>5</sup> Bilişim Sözlüğü 2007

Diğer tanımı ele alacak olursak; “Veri Ambarı, yönetimin karar verme sürecini desteklemede verinin<sup>6</sup>;

- Konu-yönelimli,
- Bütünleşik,
- Zaman değişkenli,
- Değişken olmayan bir toplamıdır.

Veri Ambarının bu kuramsal tanımına ait anahtar noktaları daha detaylı açıklamak gerekmektedir.

“**1-Konu-Yönelimli:** Veri Ambarının birinci dikkate değer karakteristiği, onun şirketin önemli konuları etrafında organize edilmiş olmasıdır. Şirketin önemli konuları etrafına yönelim, Veri Ambarı tasarımının “veri sürüslü” olmasına yol açar.<sup>7</sup>”

“Veri Ambarı; müşteriler, sağlayıcılar, ürünler vb. gibi organizasyonu yürütme konularına yönelir. Veri Ambarı, karar destek için kullanılmayacak olan veriyi barındırmaz.

Bu, Veri Ambarı ve diğer geleneksel veritabanı yönetim sistemleri arasında önemli bir farktır.<sup>8</sup>”

Veri Ambarları, kullanıcının kendi verilerini analiz etmesine yardımcı olmak üzere tasarlanmıştır. Örneğin kullanıcı, kendi şirketinin satış verileri hakkında daha fazla şey öğrenmek isteyebilir. Bunu gerçekleştirmek için, kullanıcı satışlarını ilgilendiren bir Veri Ambarı oluşturabilir. Kullanıcı bu ambarın içinde, “En çok hangi renk araç satıldı?” gibi sorulara cevap alabilecektir.

---

<sup>6</sup> Inmon, W.H., Hackathorn R. Using the Data Warehouse, s.2.

<sup>7</sup> Paul, L., George L., “Oracle8,” Data Warehousing Guide, 1999, s. 2-3.

<sup>8</sup> Srinivasan, A., Sundaram, D., Davis, J., Implementing Decision Support Systems: Methods, Techniques and Tools,Londra, 2000, s.123.

“**2-Bütünleşik:** Veri Ambarı çevresinin en önemli özelliği ve onun ikinci sınırlayıcı karakteristiği verinin çevresiyle bütünleşik olarak ambar içinde barındırılmasıdır.<sup>9</sup>”

“Bir Veri Ambarı içinde bulunan veri, organizasyon içinde ve dışında dağıtılmış halde bulunan veri nedeniyle bütünleşir, merkezileşir ve birleştirilir. Bütünleşme, konu yönelimle ilgili yakından ilgilidir. Veri Ambarları sabit bir biçim (format) içine konuları tamamen farklı kaynaklardan veri almaya ihtiyaç duyar.

Veri Ambarı verisinin bütünleşmesi farklı yollarla ortaya çıkabilir: Sürekli adlandırma kuralı, değişkenlerin sürekli ölçümü, sürekli kodlama yapıları, verinin sürekli fiziksel öznitelikleri vb.<sup>10</sup>”

“**3-Zaman Değişkenli:** Ambar içindeki verinin temel karakteristiği işlemsel çevre içinde bulunan veriden çok farklıdır.<sup>11</sup>”

“ Veri Ambarının içindekiler, zamana bağlı olarak veri akışını temsil etmektedir. Veri Ambarı içindeki veri, erişimin herhangi bir zamanında doğrudur ve geniş bir zaman aralığında bulunmaktadır. Zaman değişkeninin asıl anlamı ise Veri Ambarının veri yapıları içinde anahtar bir bileşen gibi zamanın içsel olarak hazır bulunmasıdır.<sup>12</sup>”

“**4-Değişken Olmayan:** Veri Ambarı karakteristiğinin dördüncü tanımı, onun değişken olmayışıdır. Veri Ambarında meydana gelen işlemlerin iki türünü; verinin başlangıç yüklemesi ve veri erişimi olarak tanımlayabiliriz. İşlemin normal bir parçası olarak, veri ambarında veri güncellemesi yoktur.<sup>13</sup>”

“Veri Ambarı içine giren veri, genel olarak bir daha asla çıkarılmadığı gibi daima artış göstermektedir. İlişkisel veritabanlarından farklı olarak, Veri

---

<sup>9</sup> Srinivasan, Sundaram, Davis, a.g.k., s.123.

<sup>10</sup> Inmon, Hackathorn, a.g.k., s.5-8.

<sup>11</sup> Inmon, Hackathorn, a.g.k., s.5-8.

<sup>12</sup> Srinivasan, Sundaram, Davis, a.g.k., s.123.

<sup>13</sup> Inmon, Hackathorn, a.g.k., s.10.



Ambarında depolanmış veride araya eklemeler, silmeler veya deęişiklikler yoktur.<sup>14</sup>”

Bununla birlikte zamanın belirli bir anında, belirli bir ticari işlemin durumunu sunan veritabanı yönetim sistemi Çevrimiçi Hareket İşlem (OLTP=On-Line Transaction Porcessing) ile Veri Ambarı modeli arasında bazı önemli farklar vardır.

## 1.2 Veri Ambarı Tarihi

Veri Ambarları 1980’lerin sonuna doğru ve 1990 ların başlarından itibaren farklı bir veri databanı tipi olmaya başladı. Operasyonel sistemlerle karşılanamamış bilgi yönetimi ve analizi için büyümekte olan talebi karşılamak için geliştirilmişlerdi<sup>15</sup>.

Operasyonel Sistemler bu talebi bazı nedenlerden dolayı karşılayamamıştı:

- i. Operasyonel Sistemlerin raporlama zamanlaması oldukça düşüktü.
- ii. Operasyonel Sistemlerin Veri Tabanı tasarımı bilgi analizi ve raporlama yapılabilmesi için optimize edilmiş değildi.
- iii. Organizasyonların çoğu birden fazla işletim sistemine sahipti, bu yüzden şirket içinde raporlamayı tek bir sistemden yapabilmek mümkün değildi ve Operasyonel Sistemler üzerinde rapor geliştirmek için oldukça yavaş ve pahalı olan özel yazılımlar gerektiriyordu.

Sonuç olarak, farklı veri tabanları bilgi yönetimini destekleyecek ve analiz amaçlı kullanılacak şekilde özellikli olarak tasarlanmaya başlandı. Bu Veri Ambarları farklı veri kaynaklarından, ana bilgisayarlar, mini bilgisayarlar, kişisel bilgisayarlar ve spreadsheet gibi ofis otomasyon yazılımlarını tek bir yerde bütünleştirebilecek yeteneğe sahipti.

Bu yeteneklilik kullanıcı dostu raporlama araçları ve operasyonel etkilerden bağımsız bu tarz bilgisayar sistemlerin etkisini artırarak büyümesini sağladı teknoloji ilerledikçe (daha fazla performans için daha az maliyet) ve kullanıcı

---

<sup>14</sup> Srinivasan, Sundaram, Davis, a.g.k., s.124.

<sup>15</sup> Inmon, Hackathorn, a.g.k., s.25.

ihtiyaçları arttı (daha hızlı veri yükleme süresi ve daha fazla özellikler) ve ambarları bir kaç temel aşamayı geçerek gelişti:

- i. Çevirimdışı Operasyonel Veri Tabanı: Bu aşamadaki veri ambarları operasyonel sistemin veri tabanını, operasyonel sisteminin performansını rapor oluşturtanının yükünden etkilenmeyecek bir çevrim dışı sunucuya kopyalayarak çalıştırılır.
- ii. Çevirimdışı Veri Ambarı: Gelişimin bu aşamasında Veri Ambarları düzenli zaman aralığında operasyonel sistemlerden ve bütünleşik bir rapora yönelik veri yapılarında bulunan verilerden güncellenmekteydi. (Genellikle günlük, haftalık veya aylık)
- iii. Gerçek Zamanlı Veri Ambarı: Gelişimin bu aşamasındaki Veri Ambarları operasyonel sistemler de her zaman oluşan bir işleme bağlı olarak güncelleniyordu. (Örnek: Bir Sipariş, Teslimat veya Rezervasyon işlemi sonunda)

Tümleşik Veri Ambarı: Bu aşamadaki Veri Ambarları aktiviteler veya organizasyonun günlük aktivitelerinden kullanılan operasyonel sistem tarafından geriye gelen işlemleri oluşturmak için kullanılır<sup>16</sup>.

### 1.3 Veri Ambarı Oluşturmak İçin Nedenler

Şirketler bir veri ambar oluşturmada neden para harcama yoluna giderler? Bu soruya verilecek birinci ve en önemli cevap; oluşturulan bir veri ambarının, şirketler arası rekabetler ile ilgili olarak onlara stratejik bir avantaj sağlamasıdır. Bu avantaj farklı kaynaklardan gelebilir<sup>17</sup>.

1. “Şirketlerin geniş bir alanı kapsayan verilerine erişim gücü
2. Sürekli veriye sahip olma gücü
3. Analizleri hızlı bir şekilde yerine getirebilme gücüdür.”

Şimdi bu avantajları ele alacak olursak:

---

<sup>16</sup> Inmon, Hackathorn, a.g.k., s.32.

<sup>17</sup> Microsoft Training And Certification Workbook, s.16.

1- Şirketlerin Geniş Bir Alanı Kapsayan Verilerine Erişim Gücü: Pek çok şirkette, farklı bölümlerden toplanan veriler önemli bir zorluk oluşturabilir. Ana bilgisayar veritabanlarından, çalışma tabloları ve kelime işlemci dosyalarına kadar, bölümler farklı biçimlerde veri depolayabilirler. Ambar, tüm bu verileri heterojen kaynaklardan merkezi bir yere taşır. Böylece şirkette ilk kez, konuyla ilgili tüm iş verilerine erişim imkânı tanınır.

2- Sürekli Veriye Sahip Olma Gücü: Çeşitli bölümlerden merkezi bir ambar içine veri taşıma işlemi sürekli bir şekilde gerçekleştirilir. Bir bölüme ait veriler diğer bölümlere ait verilerle karşılaştırılmadığı zaman kararların alınması zorlaşır. Verinin yararlı olabilmesi için verinin içsel olarak sürekli olması gerekir. Veri Ambarı, Data Martlar arasında sürekliliğin sağlanması ve kaynak Data Mart'a ulaşmada en iyi yoldur.

3- Analizleri Hızlı Bir Şekilde Yerine Getirebilme Gücü: Veri, ambarda ve içsel olarak sürekli olduğu zaman, analizin gerçekleştirilmesi amacıyla sorgulanabilir. Bu analiz genelde veriyi yeniden düzenleme ve eğilimleri araştırmada veri madenciliğini gerçekleştirmektedir. Veri Ambarları bireysel pazarları daha iyi anlama ve incelemede ve yaygın eğilimleri ortaya çıkarmada insanlara yardımcı olurlar.

#### **1.4 Veri Ambarı Hedefi**

"Veri Ambarı geliştirilirken hedefler organizasyonun yapısı göz önünde bulundurularak ve iş yönetiminin de fikirleri dinlenerek geliştirilmelidir.

İş tarafından gelebilecek olası sorular ve istekler aşağıdaki şekilde olabilir. Hedefler aşağıdaki sorulara ve isteklere yanıt olacaktır<sup>18</sup>.

- Bu şirketin yığınlarca verisi var fakat erişemiyoruz.
- Veri yığınının mümkün olan her yönden bölümlendirmemiz gerekmektedir.
- İş tarafının erişeceği veriler en basit ve hızlı yoldan olmalıdır.
- Önemli olan veri ne ise sadece o gösterilmelidir.

---

<sup>18</sup> Kimball, R., Ross, M., The Data Warehouse Toolkit, s.28-29.

- İki ayrı çalışanın aynı toplantıda aynı konu üzerinde farklı değerlerle rapor vermesi bizi çılına çeviriyor.
- Çalışanların bilgiyi daha çok gerçek temelli karar verme aşamalarından kullanmasını istiyoruz.”

## 1.5 Veri Ambarı Kullanıcıları

Veri Ambarı kullanıcısı – DSS analist olarak ta anılır, öncelikle bir iş adamıdır daha sonra teknik kişidir. DSS analistinin öncelikli görevi şirkette karar vermede kullanılan bilgiyi tanımlamak ve keşfetmektir<sup>19</sup>.

Veri Ambarı kullanıcıları firmalardaki tarihsel ve analiz edilebilmesi mümkün olan veriye ihtiyaç duyan departmanlardır. Veri Ambarı kullanıcıları pazarlama danışmanları gibi iş kullanıcılarıdır<sup>20</sup>.

“Bir veri ambarı, iki tip kullanıcıya hizmet eder:

- 1-Doğrudan kullanıcılar ve
- 2-Uygulama sahipleri”<sup>21</sup>.

Doğrudan kullanıcılar, işlerini gerçekleştirebilmek amacıyla veri erişimine ihtiyaç duyan pazarlama analistleri veya finansal planlayıcılarıdır. Excel veya Access gibi yalın yazılımlarla çalışarak, doğrudan ambar içindeki veriye erişebilirler. Bu kişilerin ambar içinde ne tür veri bulabileceklerini, bunların nasıl depolandıklarını ve bunlara nasıl erişebileceklerini çok iyi anlamaları gerekir. Çünkü bu grup sahip olduğu veriyi daha fazla işlemek ister.

Uygulama sahibi ise birkaç kullanıcıya sahip bir uygulamayı gerçekleştirmekle sorumludur.

## 1.6 Veri Ambarı Özellikleri ve Üstünlükleri

---

<sup>19</sup> Inmon, W.H., Building the Data Warehouse, 2005, s.20.

<sup>20</sup> Todman, C., Designing a Data Warehouse: Supporting Customer Relationship Management, s.133.

<sup>21</sup> Zeki Veri Madenciliği, Ham Veriden Altın Bilgiye Ulaşma Yöntemleri”, Bilişim, Veri Madenciliği Eğitim Seminer Notu, 2000, s.3.

Veri Ambarı kullanmanın birçok avantajı vardır, bunları aşağıdaki gibi sıralayabiliriz:

- Son kullanıcının geniş çeşitli veriye erişimini artırır.
- İş karar vericileri trend raporları elde edebilir, örneğin: belli bir alanda veya ülkede en fazla satan ürünler.

Bir Veri özellikle (CRM) Müşteri İlişkileri Yönetimi gibi ticari ve kurumsal uygulamaları daha önemli kılabilir<sup>22</sup>.

19. yüzyılın olumsuz uyarılarına rağmen, geçmiş olaylar ve eğilimlerden elde edilen bilgileri kullanmak sıradan ekonomik ve sosyal göstergelerin öngörüsü, hatta hava tahminleri, tarihsel verilerin değeri ve öneminin farkına varmamızı sağlıyor. Tarihsel bilgi saklamak veri ambarlamanın yeteneği olarak gözlenmiştir ve operasyonel sistemlerdeki tarihsel bilginin eksiliği veri ambarları geliştirmekte en önemli motivasyonlardan biri olarak kabul edilmiştir<sup>23</sup>.

Veri Ambarı kullanmanın deavantajları:

- Veri seçip çıkarma, temizleme ve yükleme oldukça zaman alan işlemlerdir
- Veri ambarı projesi kapsamında tanımlanan içerik ve değerler sonuç elde edilene kadar aktif olarak yönetilmesi gerekmektedir.
- Farklı sistemlerle uyumluluğunda sorunlar doğurabilir.
- Eğer veri ambarı web erişimine sahip ise güvenlik ciddi bir tehdit yaratabilir.
- Veri depolama tasarımı dikkatli bir şekilde göz önünde bulundurulmalı ve belki de her projenin çevresi için veri ambarı çözümünün modellenmesi gerekebilir.

## 1.7 Veri Ambarının Kavramsal Tasarımı

Veri Ambarlamada, olgu tablosu (fact table) bir iş sürecindeki ölçülerden, ölçev (metrics) veya olgularından oluşmaktadır. Olgu tablosu boyut tablolarıyla çevrilmiş bir şekilde Yıldız Şemanın merkezinde konumlandırılır .

---

<sup>22</sup> Inmon, Hackathorn, a.g.k., s.50.

<sup>23</sup> Todman, a.g.k., s.133.

Olgu tabloları genellikle boyutlu nitelikler tarafından çözümlenmiş bağımsız değişkenler gibi rol oynayan ilave değerler sağlar. Olgu tabloları sıklıkla tanecikleriyle (grain) tanımlanır. Bir olgu tablosunun taneciği tanımlanabilen en küçük veri ögesidir. (atomic level)

Satış Olgu tablosunun granularity level'i "Günlük, Ürün Başına, Mağaza Başına düşen Satış Hacmi olabilir". Bu olgu tablosundaki her kayıt günlük, ürün başına ve mağaza başına tanımlanmış tekil kayıtlardır. Bu olgu tablosunun diğer kayıtları Üyeler olabilir. (Konum/Bölge) Fakat bu veriler olgu kayıtlarının tekilliğine hiçbir şey eklemes. Bu "affiliate dimensions" katılan diğer boyutlar bağımsız olgulara ilave dilimler eklenmesine izin vermektedir fakat genellikle daha üst seviyede anlayış sağlanabiliyorlar. (Bölge bir kaç mağazadan oluşmaktadır)

Eğer iş süreci SATIŞLAR ise, ilişkin olgu tablosu tipik olarak aşağıdaki ham olguları ve birleşimleri içerebilir:

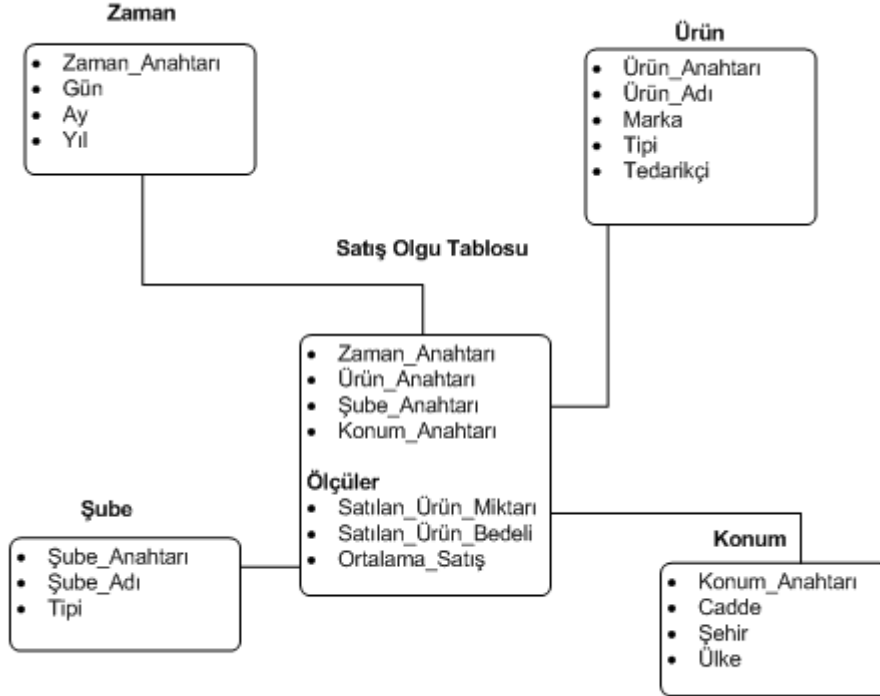
12.000 YTL "İSTANBUL daki mağazadan 15 Ocak 2007 yapılan satış"  
34.000 YTL "ANKARA daki mağazadan 16 Ocak 2007 yapılan satış"  
16.000 YTL "BURSA daki mağazadan 16 Ocak 2007 yapılan satış"  
22.000 YTL "İSTANBUL daki mağazadan 17 Ocak 2007 yapılan satış"  
32.000 YTL "ANKARA daki mağazadan 17 Ocak 2007 yapılan satış"  
18.000 YTL "BURSA daki mağazadan 18 Ocak 2007 yapılan satış"

Veri Ambarlarını modelleme de kullanılan boyutlar ve ölçüler.

### **1.7.1 Yıldız Şema (Star Schema):**

Bir olgu tablosu birçok boyutlu tabloların ortasında bağlanmış bir şekilde gözüküyor.

Şekil 1: Yıldız Şema<sup>24</sup>

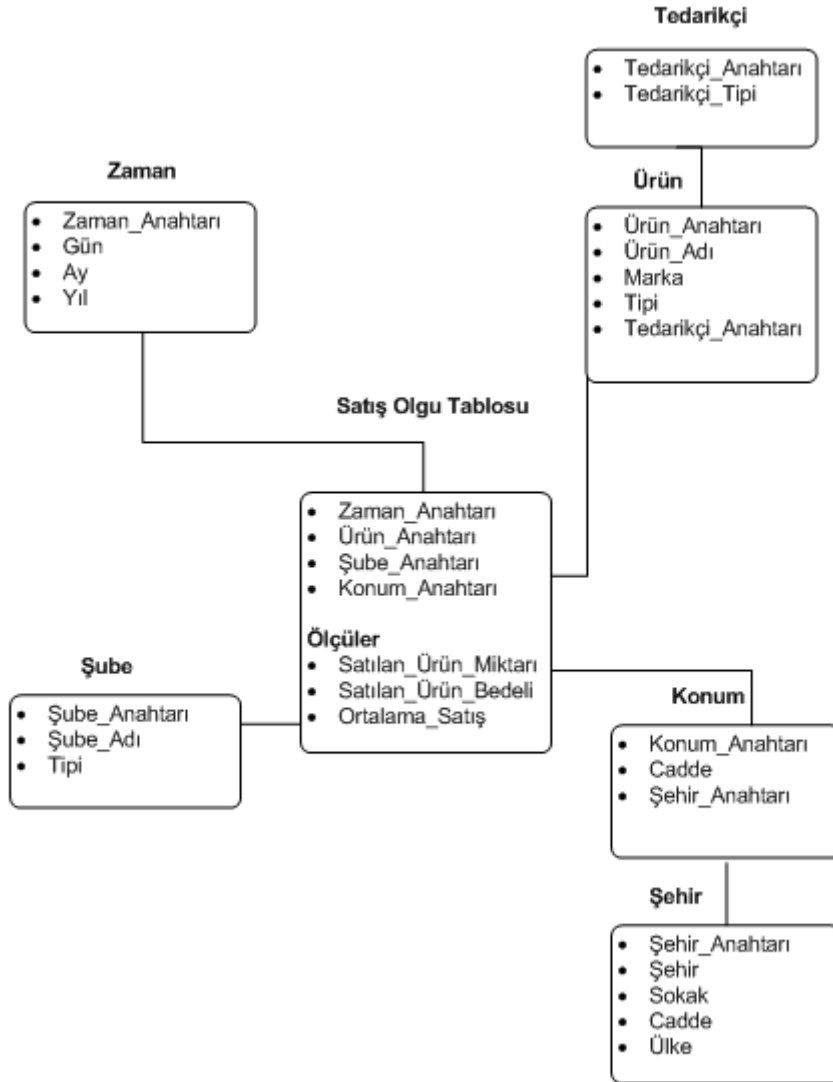


<sup>24</sup> Harinarayan, V., Rajaraman, A., Ullman, J.D., Implementing data cubes efficiently. 1996, s.205.

## 1.7.2 Kar tanesi Şeması (Snowflake Schema)

Yıldız Şemanın boyutlu hiyerarşik yapının daha küçük boyutlu tablolara normalize edilmesi, kar tanesine benzer yapıya dönüştürülerek ayrıştırılması işlemidir.

Şekil 2: Kar tanesi Şeması<sup>25</sup>



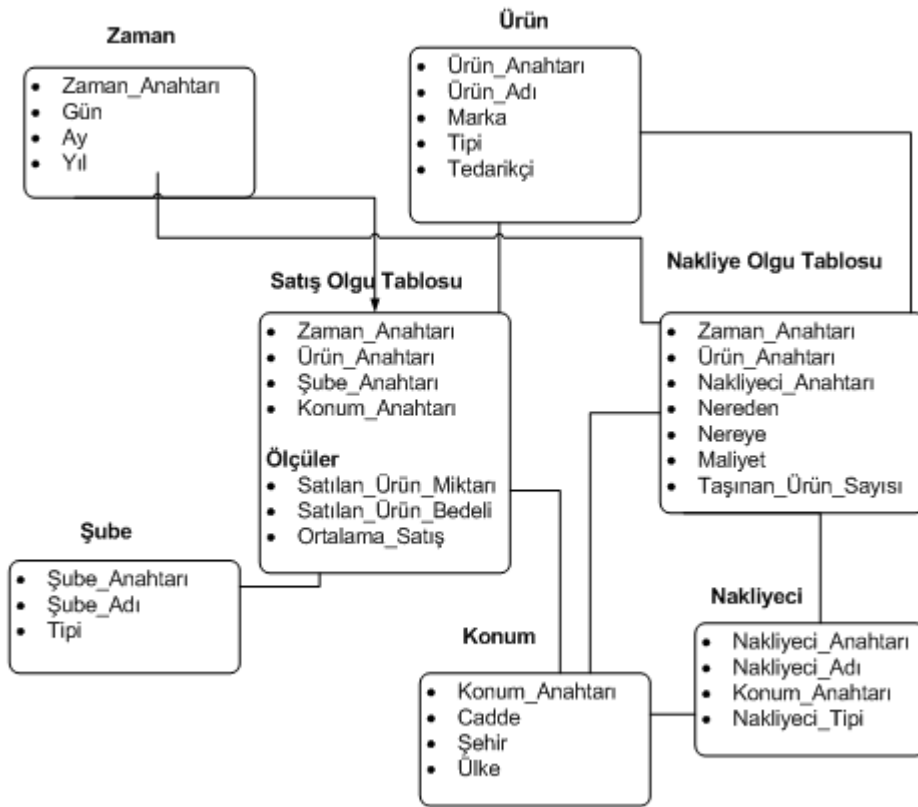
<sup>25</sup> Harinarayan, Rajaraman, Ullman., a.g.m., s.206.



### 1.7.3 Takımyıldız Olgusu (Fact Constellations) veya Galaksi Şeması (Galaxy Schema)

Çoklu olgu tabloları yıldızlar topluluğu gibi gözükken boyutlu tablolara paylaştırılır, böylece Galaksi Şeması veya Takımyıldız Olgusu olarak adlandırılır.

Şekil 3: Takımyıldız Olgusu<sup>26</sup>



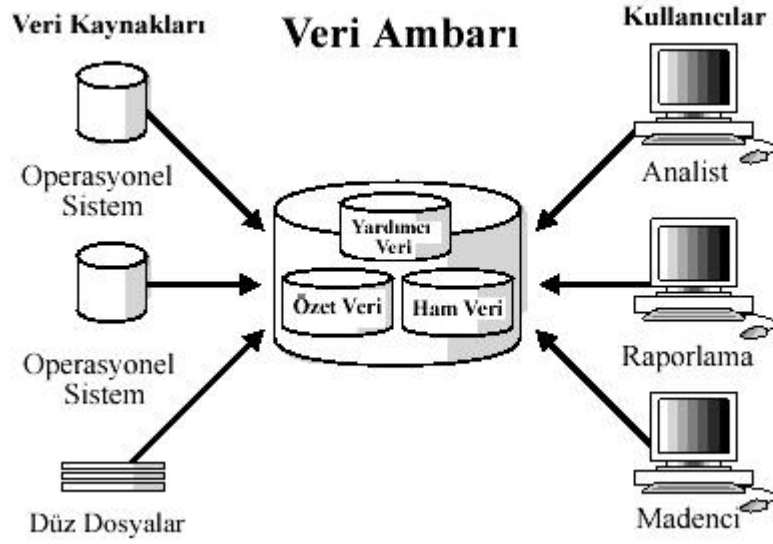
<sup>26</sup> Harinarayan, Rajaraman, Ullman., a.g.m., s.206.

## 1.8 Ambar Mimarisi

### 1.8.1 Veri Ambarı Mimarisi (Temel)

Şekil 4 de basit bir Veri Ambarı Mimarisini temsil etmektedir. Son kullanıcı bir kaç kaynak sistemden türetilen veriye veri ambarı vasıtasıyla doğrudan erişiyor.

Şekil 4: Veri Ambarı Mimarisi<sup>27</sup>

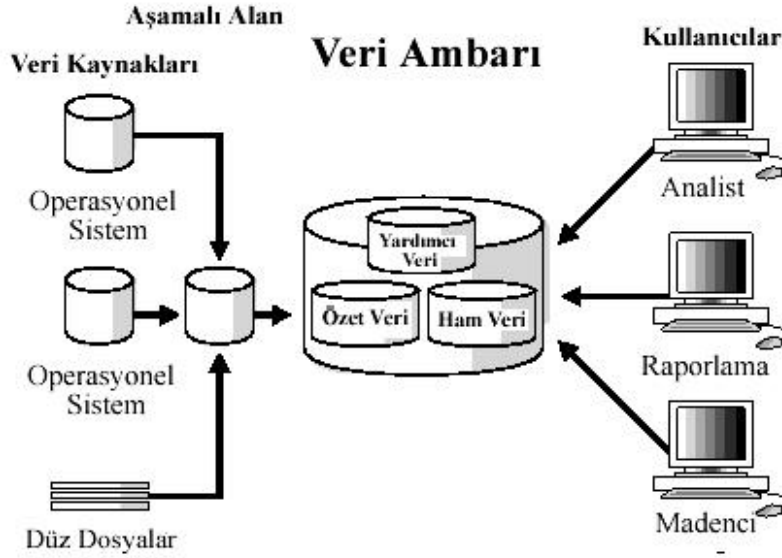


<sup>27</sup> Paul, L., George L., a.g.k., s.1-5.

## 1.8.2 Veri Ambarı Mimarisi (Aşamalı Alan – Staging Area)

Operasyonel Veriyi, Veri Ambarına almadan önce temizlemek ve işlemek gerekmektedir. Birçok Veri Ambarı Aşamalı Alan kullanmasına rağmen bunu program yardımıyla yapabilirsiniz Aşamalı Alan özetler oluşturmayı ve genel Veri Ambarı yönetimini basitleştirir.

Şekil 5: Aşamalı Alanlı Veri Ambarı Mimarisi<sup>28</sup>

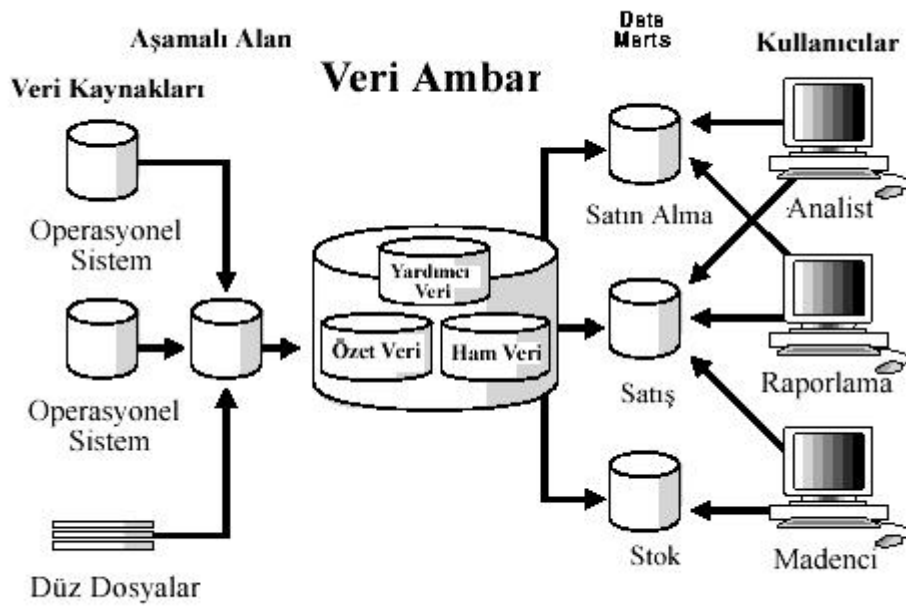


<sup>28</sup> Paul, L., George L., a.g.k., s.1-5.

### 1.8.3 Veri Ambarı Mimarisi (Aşamalı Alan ve Data Mart)

Şekil 6 mimari ortak olmasına rağmen, Organizasyonunuzun içinde Veri Ambarınızın mimarisini farklı gruplara özelleştirmek isteyebilirsiniz. Bunu özel iş kademeleri için tasarlanmış Data Martlar ekleyebilirsiniz, Şekil 6 da satın alma, satış ve ayrıştırlmış stok tasvir etmektedir. Bu örnekte, tarihsel finansal analist satın alma ve satış verisini analiz edebilir.

Şekil 6: Aşamalı Alan ve Data Martlı Veri Ambarı Mimarisi<sup>29</sup>



<sup>29</sup> Paul, L., George L., a.g.k., s.1-5.

## 1.9 Veri Ambarı Sunucu İstemci Araçları ve Yardımcı Uygulamalar

Bir Veri Ambarını yönetmede gerekli görevleri şu şekilde sıralayabiliriz<sup>30</sup>:

- 1-Seçip çıkarma (Extraction)
- 2-Taşıma (Transportation)
- 3-Dönüştürme (Transformation)
- 4-Yükleme ve Yenileme (Loading and Refreshing)

Ambar çevresi yönetimindeki ilk üç görevi kısaca ETL şeklinde tanımlamak mümkündür. Kullanıcı düzenli bir şekilde kendi veri ambarını yüklemeye ihtiyaç duyar ve böylece sağlanan iş analizinin amacına hizmet edebilir. Bununla birlikte, bir veya daha fazla işletimsel sistemlerinden gelen veri seçip çıkarılmaya ve ambar içine kopyalanmaya da ihtiyaç duyabilir. Okuma ve veri hazırlama işlemi nispeten zordur ve düzenli bir temel üzerinde gerçekleştirilmesine ihtiyaç duyar.

1- Seçip Çıkarma (Extraction): Seçip çıkarma bir veritabanından bir dosya içine veya bir ağ bağlantısı üzerine veri kopyalama işlemidir. Bu ETL işleminin birinci adımıdır. Veri, kaynak sistem veya sistemlerden seçip çıkartılmalıdır böylece, bu veri sonradan Veri Ambarı içine dönüştürülebilir ve yüklenebilir.

Bir Veri Ambarı için kaynak sistemler tipik olarak hareket-işlem veritabanı uygulamalarıdır. Örneğin, bir satış-analizi veri ambarı için kaynak sistemlerden biri, mevcut sipariş etkinliklerinin tümünü kayıt eden sipariş-giriş sistemi olabilir.

Veri seçip çıkarmada kullanılan teknikler iki sınıfa ayrılır:

- -Bir işletim sisteminden veri seçip çıkarma ve bir dosya içine yerleştirme teknikleri. Örnek olarak, veriyi boşa alma ve dışarı aktarım.
- -Bir işletimsel sistemden veri seçip çıkarma ve doğrudan hedef veritabanı içinde taşıma teknikleri. Örnek olarak, ağ geçitleri ve dağıtılmış sorgular.

---

<sup>30</sup> Paul, L., George L., a.g.k., s.10-12.

2-Taşıma (Transportation): Taşıma, tam olarak, bir sistemden diğer bir sisteme verinin taşınması işidir. Bir Veri Ambarı çevresinde, taşıma amaçlı en genel gereksinimler bir kaynak sistemden bir safhalama veritabanına veya bir veri ambarı veritabanına; bir safhalama veritabanından bir veri ambarına; veya bir veri ambarından bir data mart'a verinin taşınmasıdır.

3-Dönüştürme (Transformation): Veri dönüşümleri işlem zamanı koşullarında, çoğunlukla en karmaşık ve ETL işleminin en pahalı kısmıdır.

4-Yükleme ve Yenileme (Loading and Refreshing): Seçip çıkarma ve dönüştürmenin ardından, ETL işleminin son adımı, fiziksel olarak yeni, temiz veriyi üretim veri ambarı şeması içine yerleştirmektir.

### **1.10 Ayır Veri Ambarı Oluşturma Nedeni**

Bilindiği üzere Veri Ambarı, Çevrimiçi Hareket İşlem (OLTP) sisteminden ayrıdır. Bu ayırım için dört neden vardır:

“1-Performans: OLTP sistemlerinde veri, karar destek sorunlarından daha çok işlemsele yönelir. Sonuç olarak basit sorguların OLTP sistemi içinde büyük karışıklıklara neden olduğu görülür.

2-Veri Erişimi: Organizasyonlar çoğunlukla farklı OLTP işlevlerine hizmet veren çoklu veritabanlarını paraca destekler. Şirketin tüm verisiyle bütünleşen Veri Ambarı tüm bu veri kaynaklarını birleştirir ve dışsal veri kaynaklarını ekler. Karar destek uygulamaları için veri, bu çoklu kaynakları kullanır. Tipik Veri Ambarı kullanıcısı verinin ambarlandığı yere önem vermez.

3-Veri Biçimleri (Format): Ambar içindeki veri, özetlenmiş veriyi ve OLTP sistemleri içinde tutulmayan zaman-tabanlı veriyi içerir. Çünkü ambar içindeki veri bütünleşiktir. Bilgi tek bir standart biçim içinde tutulur.

“4-Veri Kalitesi: Ambar içindeki veri temiz, geçerliliği denetlenmiş ve gereği gibi kümelenmiştir”<sup>31</sup>.

---

<sup>31</sup> Gray, Paul, Watson, J., a.g.k, Decision Support In The Data Warehouse, 1998, s.19.

### **1.11 Data Mart (Küçük Veri Ambarı)**

Bir Data Mart bir Veri Ambarının özelleştirilmiş sürümüdür. Veri Ambarları gibi Data Martlar da iş tarafına tarihsel eğilimler ve deneyimlerin analizinden yardımcı olacak operasyonel verinin bir enstantanesini saklamaktadır. Data Mart oluşturmaktaki anahtar farklılık önceden tanımlanmış belli bir ihtiyaca yönelik bir grup bilgi ve seçilmiş veri yapılandırılmasıdır. Data Mart yapılandırılmasıyla ilgili veriye kolay erişim vurgulamaktadır. Data Martların kavramsal tasarımları da Veri Ambarları gibi Yıldız Şema, Kar tanesi Şeması ve Takımyıldızı şemalarından oluşmaktadır.

### **1.12 Data Mart Oluşturmak İçin Nedenler**

Veri Ambarlarının yüksek maliyeti, ambarların büyük şirketler tarafından kullanımını sınırlamaktadır. Pek çok şirket tarafından kullanılmaya başlanan alternatif ise; daha düşük maliyetli, Veri Ambarının küçük sunumu olan Data Martları oluşturmaktır. Nedenlerin listesini aşağıdaki şekilde sıralayabiliriz<sup>32</sup>:

- İhtiyaç olan veriye sürekli ve kolay erişim imkânı
- Bir grup kullanıcıdan tarafından kolektif bakış oluşturabilmek
- Oluşturulması kolaydır
- Veri Ambarı uygulamaya göre maliyeti daha düşüktür
- Veri Ambarına göre Potansiyel kullanıcılar daha net bir şekilde tanımlanabilir

### **1.13 Data Mart ve Veri Ambarı Farkları**

Bir Data Mart kendine özgün, konu yönelimli, belli bir grup kullanıcıların özgün sorularına cevap olarak tasarlanmış bir depodur. Böylece bir organizasyon pazarlama, satış, operasyon bölümleri için ihtiyaçlarını karşılamak için birden fazla

---

<sup>32</sup> <sup>32</sup> Paul, L., George L., a.g.k., s.20-22.

Data Mart kullanılabilir. Bir Data Mart genellikle olgu tablosu ve çok boyutlu tablolardan yapılmış bir boyutlu yıldız şemasını model alır (OLAP küpleri) <sup>33</sup>.

Diğer taraftan, bir DW tek organizasyon için birçok veya tüm konularda veri içeren kurumsal çapta bir veri deposudur. Veri Ambarı bütün olguların ve atomik seviyedeki (atomic level) boyutlardaki verilerin yetkili deposudur. (Data Martlarda da geçerli olabilir)

---

<sup>33</sup> Inmon, Hackathorn, a.g.k., s.74.



## II.BÖLÜM

### OTOMOTİV SEKTÖRÜ

Bu bölümde Dünya ve Türkiye Otomotiv Sanayi, Doğu Otomotiv Servis Tic. A.Ş. Tarihçesi ve Satış Sonrası Hizmetleri doğrultusunda Servis Noktalarının da yapılan Operasyonel İşlemler anlatılacaktır..

#### 2.1. Dünya Otomotiv Sanayii

Dünya otomobil sanayinin gelişmesinde ABD ve Japon firmalarının önemli rolleri olmuştur. Özellikle, ABD otomotiv sanayinin öncü kuruluşlarından "Ford" ve Dünya otomobil piyasasında önemli bir yeri olan Japonya'nın "Toyota" şirketi, otomobil sektöründe ekol teşkil edecek mahiyette karakteristik özelliklere sahiptir.<sup>34</sup>

Atölyede el marifetiyle yapılmış olan ilk Ford otomobil, ikinci aşamada geliştirilen bir modelin bölgesel ihtiyaçlara göre küçük çaplı değişikliklerin yapıldığı montaj tesislerinde (assembly-line methods) üretilmeye başlanmasıyla üretim ve pazarlamada genişleme sürecine girmiştir. Bu üretim tarzı, Amerikan otomobil yapımcılarının uluslararası piyasalara doğrudan ihracat olarak ilk açılma hareketidir. Üçüncü aşamada üretimin pazara yaklaştırılması (transplant) üretim-ihracat modeli geliştirilerek, Avrupa ve Latin Amerika'ya doğrudan yatırımlar yapılmıştır. Bu anlayış Ford otomotiv sanayinin dünya ölçeğinde üretim yapan bir konuma ulaşmasını sağlamıştır. Bu üretim ve ihracat modeli, daha sonra Japon şirketlerinin diğer piyasalara girmede başvurdukları bir yöntem olmuştur.

Toyota firması, teknoloji ve organizasyon metodlarında kaydedilen yenilikleri, otomobil yapım sanayiinde ve pazarlamada süratle uygulayarak, üretim yapısında esneklik sağlamış, geniş ürün çeşitliliğine ulaşma başarısını göstermiştir. Firma, sahip olduğu teknoloji ve esneklik kabiliyetiyle doğal çevreye uygun, yakıtta tasarruf sağlayıcı, yeni tüketici talebi yaratacak mahiyette geliştirdiği otomobillerle piyasada önemli bir paya sahip olmuştur. Bu süreçte kalite kontrolü, tüketici tercihlerinin takip edilerek değerlendirilmesi (feedback), bölgesel şartlara uygun ürün geliştirilmesi vazgeçilmez unsurlar olmuştur.

---

<sup>34</sup> [www.dtm.gov.tr/dtmadmin/upload/EAD/IstatistikDb/dun1.doc](http://www.dtm.gov.tr/dtmadmin/upload/EAD/IstatistikDb/dun1.doc)

Diğer taraftan, ABD firmalarınca geliştirilmiş olan transplant üretim modeli (üretim pazara yakın noktalara kaydırılması) 1970'li yıllardan itibaren Japon otomobil yapımcısı firmalar tarafından daha da geliştirilerek, ABD başta olmak üzere, AB ülkeleri ve diğer ülkelerde yaygın olarak kullanılmıştır.

Otomobil sektörü, dünya genelinde sahip olduğu güçlü tüketici talebiyle dinamik bir yapı arz etmektedir. Sektör; kullanılan teknoloji, üretim miktarı, istihdam hacmi, ürün çeşitliliği ve ticaret hacmi bakımından global etkiye sahip bir konuma gelmiştir. Ancak, gerek üretimin miktar ve kalitesi, gerekse ticaret ve yatırımda piyasa şartları ve tüketici esnekliklerinin değerlendirilmesinde görülen bölgesel farklılıklar, sektörün henüz dünya ölçeğinde homojen bir yapıya ulaşamadığının göstermektedir.

## **2.2. Türkiye Otomotiv Sanayii**

Türklerin otomobille ilk tanışması yirminci yüzyılın başlarında olmuştur. I.Dünya Savaşından sonra, İstanbul'un işgal altında olduğu yıllarında "American Foreign Trade" şirketi vasıtası ile ABD'nin Ford ve Chevrolet otomobil ve kamyonları; Torino'ya bağlı olarak İstanbul'da faaliyet gösteren Özel bir büro vasıtası ile de İtalyan Fiat Marka otomobiller piyasaya girmiştir<sup>35</sup>.

Diğer taraftan, 1929 yılında Ford Motor Company İstanbul'da ilk montaj denemesine başlamıştır. Otomobil, traktör ve kamyon üretmek üzere kurulan montaj fabrikasında yapılan üretimin bir kısmının Sovyetler Birliğine ihraç edilmesi öngörülmüştür. 450 işçi çalıştıran fabrika modern sayılabilecek teknolojik imkanlara sahipti. Tesiste, günlük 48 adet kamyon ve otomobil yapım kapasitesine ulaşılmış; ancak, 1930'larda yaşanan Dünya ekonomik krizinin olumsuz etkileri nedeniyle, hedeflenen ihracat gerçekleştirilememiştir. Bu ve diğer sebeplerden dolayı 1934 yılında fabrikada üretim durdurulmuş ve ilk montaj üretim denemesi böylece başarısızlıkla sonuçlanmıştır. Bu başarısızlık, Ford Motor Company yetkililerinin daha sonraki yıllarda Türkiye'de üretim yapma konusunda yapılan teklifleri değerlendirmekte tereddüt etmelerine sebep olmuştur.

---

<sup>35</sup> <http://www.dtm.gov.tr/dtmadmin/upload/EAD/IstatistikDb/turkiyeoto.doc>

II. Dünya Savaşının sona ermesiyle otomobil üretiminde ilk teşebbüsü "Koç Ticaret Şirketi", "Ford Motor Company"nin Türkiye Genel Temsilciliğini alarak başlatmıştır. 1950'li yılların başlarında, Koç Ticaret Şirketi, Ford Motor Company ile Türkiye'de ortak üretim yapılması konusunda temasa geçmiştir. Ford Motor Company'nin sahibi Henry Ford II'nin konu üzerinde çekingen davranması üzerine, devrin başbakanı Adnan Menderes Henry Ford II'ye bir mektup yazarak, Koç Ticaret Şirketi ile ortak üretim konusunda teşvik etmiştir. Bunu takiben, 1956 yılında Koç Ticaret A.Ş.' ye "Montaj Hakkına Sahip Bayi" "Dealer Assembler" statüsünde imtiyaz verilmiştir. Elde edilen bu imtiyaza bağlı olarak kurulan montaj fabrikası günde 8 adet kamyon, 4 adet binek otomobili üretim kapasitesi ile faaliyetine başlamıştır.

Başka bir önemli teşebbüs %100 yerli otomobil yapılması yönündeki düşünceye dayanan ve 1960 sonrası yönetiminin de desteği ile üretimine başlanmış olan "Devrim Otomobili" yapılması teşebbüsüdür. Ancak, üretilen ilk araçların denemesinin başarısızlıkla sonuçlanması bu teşebbüsünde kapatılmasına sebep olmuştur.

Bugün Ülkemizde 20 adet firma, otomotiv sanayi alanında üretim yapmaktadır. Bu firmaların 2005 yılı itibariyle 879.452 adet otomobil olan üretim adedi, 2006 yılında ise 987.580 adet otomobil seviyesinde gerçekleşmiştir<sup>36</sup>.

Türkiye'de otomotiv sanayi üretimi, 1970 yılına kadar kayda değer bir gelişme gösterememiştir. 1963 yılında 30 adet olan toplam otomobil üretimi, 1970 yılında 3.660, 1980 yılında 31.529, 1990 yılında 167.556 ve 1997 yılında 242.780 ve 1998 yılında 239.937 adet olarak gerçekleşmiştir. Otomotiv sanayinde en önemli üretim artışı 1992 ve 1993 yıllarında gerçekleşmiştir. Bu yıllarda kapasite kullanım oranları %77'ye, otomobil üretimi de sırasıyla 265.245 ve 348.095 adet seviyelerine ulaşmıştır.

Ancak, gerek 1994 ekonomik krizinin üretim üzerindeki olumsuz etkileri gerekse 1996 yılında Gümrük Birliğine girilmesi ile anlaşma gereği, AB menşeli otomobillerin ithalatından alınan gümrük vergisinin sıfırlanması ve Toplu Konut

---

<sup>36</sup> <http://www.osd.org.tr/basin2006.doc>

Fonunun (TKF) kaldırılması otomotiv üretimini olumsuz yönde etkilemiştir. 1996 yılından itibaren çeşitli ölçek, model ve fiyatta araçların piyasaya girmesi kalite ve model çeşidi olarak AB otomotiv sanayi ile rekabete henüz hazır olmayan yerli üretimi olumsuz yönde etkilenmiş ve iç talebin karşılanmasında ithalatın payı artmaya başlamıştır. 1993 yılında 548.404 adet olan otomobil ve ticari araç talebinin %25'i ithalatla karşılanırken, 1994 yılında iç piyasada oluşan talep daralmasına paralel olarak, otomotiv sektöründe de iç talep %50 oranında azalmıştır. 1994 ve 1995 yıllarında ithalatta %60 oranında düşüşler görülmüştür. 1996 yılından itibaren talep artışına paralel olarak ithalat tekrar artış eğilimine girmiştir. Bu devrede yüksek stokla çalışmanın ekonomik olmadığı da dikkate alınarak, yerli üreticiler için marjinal karlılık, üretimin azaltılması noktasında sağlanmıştır. Bu düşünceye paralel olarak 1992 ve 1993 yıllarında %77 seviyesine ulaşan kapasite kullanım oranları 1994 ve 1995 yıllarında %46, 1996 ve 1997 yıllarında da %49 ve 1998 yılında %50 seviyelerine düşerek, üretimde önemli düşüşler yaşanmıştır.

Otomotiv sektöründe, 1993 yılında ulaşılmış olan %77 oranındaki kapasite kullanım oranı hariç, 1992-1998 yılları arasında, %70 olan dünya kapasite kullanım ortalamasının gerisinde kalmıştır.

Ancak, rekabette fiyat kadar ürün farklılaştırmasının da önemli bir faktör oluşturduğu düşüncesinden hareketle, yerli üreticiler bazı yeni model otomobilleri Türkiye'de üretmeye ya da ana firmalar tarafından üretilen bazı modelleri ithal ederek rekabet güçlerini koruma çabası içine girmişlerdir.

### **2.3. Doğu Otomotiv Servis Ticaret A.Ş. Tarihçe**

Türkiye'nin lider otomotiv ithalatçısı ve en büyük distribütörlerinden olan Doğu Otomotiv; binek araç, hafif ticari araç, ağır vasıta ile endüstriyel motorlar segmentinde 13 markanın 70'e yakın modelini Türkiye otomotiv pazarına sürmektedir<sup>37</sup>.

Sektörünün lider kuruluşlarından olan Doğu Otomotiv Servis ve Ticaret A.Ş. (Doğu Otomotiv), 1994 yılında VW markası ithalat ve distribütörlüğüne

---

<sup>37</sup> [http://www.dogusotomotiv.com.tr/icerik/108/364/dogus\\_otomotiv\\_hakkinda](http://www.dogusotomotiv.com.tr/icerik/108/364/dogus_otomotiv_hakkinda)

başlamıştır ve bugün VW AG'nin yüksek satış oranına sahip markalarını bünyesinde bulundurmaktadır.

Doğuş Otomotiv dünyayla eş zamanlı olarak Türk otomobil pazarına sunduğu Volkswagen, Audi, Porsche, SEAT, Skoda, Scania ve Krone markalarının satış-servis-yedek parça hizmetini tüm Türkiye'ye yayılmış olan yaklaşık 249 adet satış noktası ile sağlamaktadır. Doğuş Otomotiv Türkiye'de temsil ettiği VW Binek, VW Ticari, Audi, Porsche, Seat ve Skoda markaları ile Otomotiv Distribütörleri Derneği toptan pazar bilgilerine göre ithal otomobil pazarında liderdir.

Doğuş Otomotiv, 19 Mart 2004 tarihinde grup şirketlerimizden Audi ve Porsche markalarının ithalat ve distribütörlüğünü yürüten Doğuş Motor Anonim şirketi, otomotiv şirketleri iştirak portföyünü yöneten Doğuş Otomotiv Holding Anonim Şirketi, Scania ve Krone ithalatçısı ve distribütörlüğünü yürüten Doğuş Ağır Vasıta Anonim Şirketi ve yedek parça ile lojistik destek fonksiyonlarını yürüten Genpar Otomotiv Anonim Şirketi ile birleşmesini tamamlamıştır.

Bu birleşme ile VW, Audi, Porsche, Scania, Krone markalarının distribütörlüğünü üstlenmesinin yanı sıra tüm bu markalara ilişkin yedek parça ve lojistik faaliyetlerini de merkez ofisi ve Gebze tesislerinde yürütmeye başlamıştır. Şirketimiz ayrıca ortaklıkları ve iştirakleri vasıtasıyla, perakende araç satış ve servis hizmetlerini, Seat ve Skoda markasının satış ve dağıtımını, ayrıca araç kredilendirme ve filo satışı ile ikinci el araç satış faaliyetlerini sürdürmektedir.

Doğuş Otomotiv'in üç ana faaliyet konusu bulunmaktadır: İthalat ve distribütörlük, üretim ve perakendecilik, satış sonrası hizmetler, otomotivle ilgili diğer hizmetler.

Doğuş Otomotiv iştiraklerinden Doğuş Oto Pazarlama A.Ş. vasıtasıyla, VW, Audi, Porsche, Seat ve Skoda marka binek araç ve hafif ticari araçların perakende ticareti yapılmaktadır.

Müşterilerine, Türkiye'de 249 satış noktasından ithal ettiği tüm araçlar için hizmet sunan Doğuş Otomotiv; son yıllarda büyüme stratejisine uygun olarak rekabet gücünü artırmak ve gelir kaynaklarını çeşitlendirmek amacıyla şirket ana işine çeşitli otomotivle ilgili hizmetleri eklemiştir. Bunlar; tüketici finansmanı (vdf

aracılığı ile), yedek parça ve aksesuar ticareti, sigorta hizmetleri, ikinci el araç ticareti, motor sporları ve filo kiralama olarak özetlenebilir.

Faaliyetlerini en üst düzeyde katma değer yaratma anlayışı ile gerçekleştiren Doğu Otomotiv; distribütörlüğünü yaptığı tüm markalarla ilişkilerini derinleştirerek büyümeyi hedeflemektedir. Şirket, bu anlayış çerçevesinde Krone üretimi için bir iyi niyet anlaşması ile düğmeye basmıştır. Aynı şekilde; VW AG ile ortaklığın güçlendirilmesi, genişletilmesi yönündeki planları doğrultusunda; VW Ticari Araç markasının Mısır'daki satış ve satış sonrası hizmetlerini üstlenmiş ve bununla birlikte, VW Deniz Motorlarının Türkiye temsilciliğini üstlenmiştir. Son olarak, Doğu Otomotiv Avrupa'da kamyon üstü damper ve yarı römork damper sektörlerinde faaliyet gösteren Meiller anlaşmıştır.

#### **2.4. Servis İş Modeli**

Doğu Otomotiv Servis Ticaret A.Ş. Audi, Volkswagen, SEAT, Porsche, Scania, Bentley ve Lamborghini marka araçları Türkiye Distribütörüdür. Bu marka araçların müşterilerine Türkiye'de Servis hizmeti vermektedir. Müşteriler Servis hizmeti almadan önce mutlaka randevu almak zorundadır. Randevu alındıktan sonra Serviste Arac üzerinde yapılan işlemlere İş Emri denilmektedir. İş modelinde belirtilen ana varlıklar Müşteri, Araç, Randevu, İş Emri varlıklarıdır.

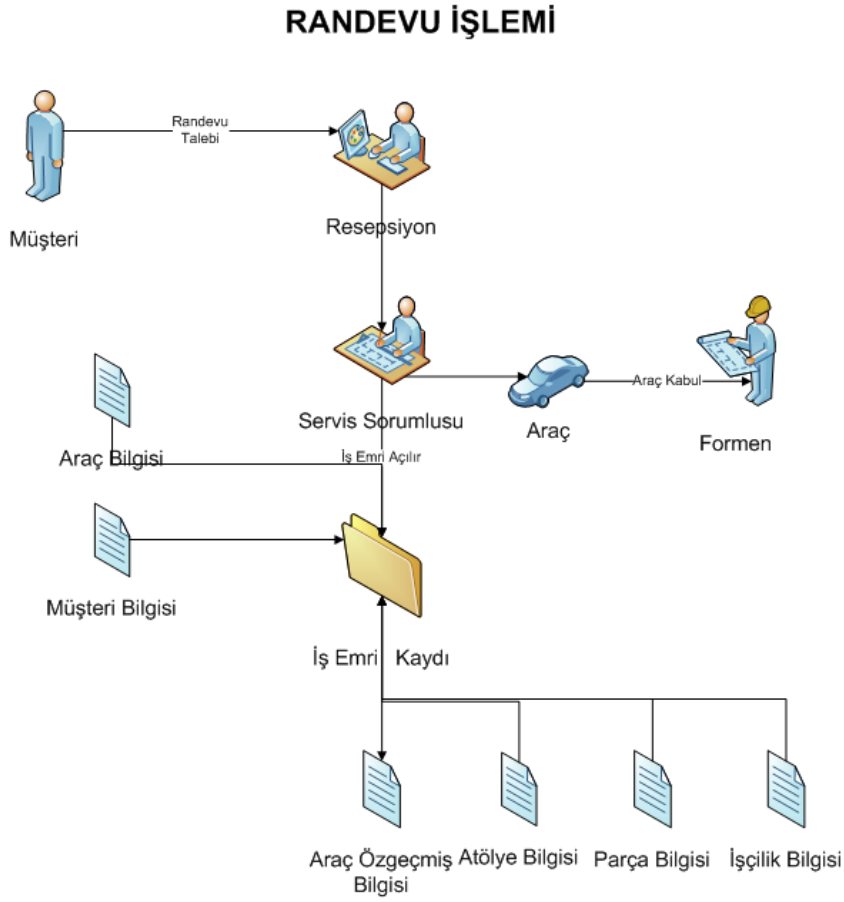
Veri Ambarı oluşturmadan önce cevaplanması gereken birkaç sorumuz var.

1. Veri Ambarı nerede barındırılacak?
2. Hangi teknolojiler kullanılarak geliştirilecek?
3. Veri Ambarı nasıl doldurulacak?

Bu tarz problemler birçok organizasyon tarafından Veri Ambarı oluşturma aşamasından önce karşılaşılabilecek problemlerdir.

Çalışmamızda; Doğu Otomotiv Bünyesinde geliştirmekte olduğumuz SSH Veri Ambarı için Doğu Otomotiv Ağı üzerinde çalışanların erişebileceği güvenliği bir alanda yeni bir MS SQL Server kurulu bir sunucu tanımlanmıştır. Veri Ambarının beslenmesi ETL işlemi ile periyodik olarak her gece saat 00:00 den sonra otomatik işlemlerle olacaktır. ETL işlemine daha sonra değineceğiz.

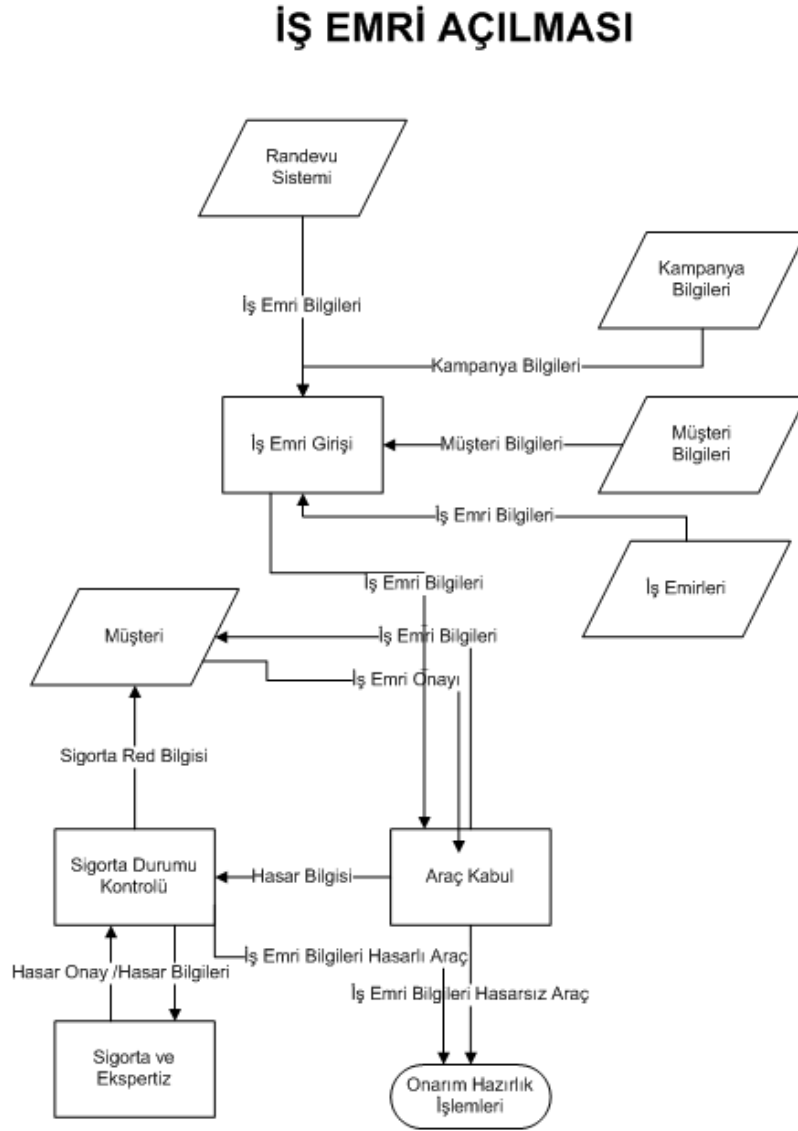
## 2.4.1. Randevu İşlemi



**Şekil 7: Servis Randevusu İş Akışı**

Doğuş Otomotiv bünyesinde verilen servis hizmeti için müşteri servise gelmeden önce randevu almak zorundadır. Servis randevusu telefon, internet veya doğrudan servis danışmanı ile konuşularak alınabilir. Randevu işleminde müşteri, servis noktasına geldikten sonra aracı güvenlik tarafından teslim alınır, daha sonra resepsiyona randevu talebini iletir, resepsiyon müşteriyi ilgili servis sorumlusuna yönlendirir. Servis Sorumlusu randevu talebiyle ilgili iş emri kaydı açar, iş emrinin içerisinde Müşteri Bilgisi ve Araç Bilgisi de yer almaktadır. Aynı zamanda Servis Sorumlusu müşterinin aracını konuyla ilgili bölümün formenine yönlendirmektedir. Formen İş Emri kaydı ile Araç Özgeçmiş Bilgisi, Atölye, Parça ve İşçilik bilgilerini de düzenlemeye başlar.

## 2.4.2. İş Emri Açılması



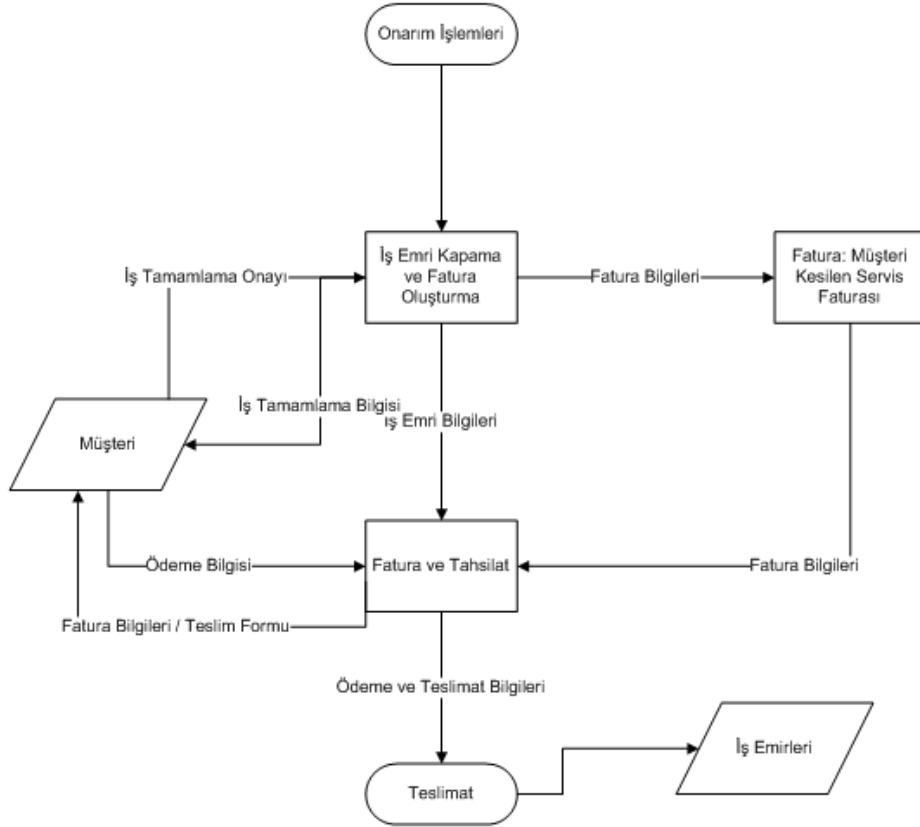
**Şekil 8: Servis Randevusu İş Emri Akışı**

İş Emri, Servis Danışmanı tarafından açılır. İş Emrinin Açılması sırasında izlenilecek yol ve adımlar Şekil 8 de gösterildiği gibi işlemektedir.



### 2.4.3. İş Emri Tamamlanması

## İŞ EMRİ TAMAMLAMA



Şekil 9: İş Emri Tamamlama

İş Emrinin tamamlanması işlemi; Onarım İşlemleri bittikten sonra Ödeme ve Faturalama işlemleri tamamlanarak Servis Danışmanı tarafından İş Emri Kaydı kapatılarak yapılır.

### III. BÖLÜM VERİ AMBARI TASARIMI

#### 3.1. Veri Ambarı Kavramsal Model Tasarımı

Doğuş Otomotiv SSH Veri Ambarı yığınlar halindeki veriye erişebilmek için gerekli olan bilgi mimarisini desteklemektedir. OLTP sistemden Data Warehouse tasarımına geçilirken önemli tablolar olan Müşteri, Araç, Kampanya, Kurum, Personel, Randevu ve İş Emri Tablolarında denormalizasyon yapılmıştır. Doğuş Otomotiv SSH Veri Ambarı Kavramsal Modeli Kartanesi Şema üzerine kurulmuştur. Modelimiz, Boyut ve Boyut tablolarının gereksinimleri doğrultusunda oluşturulmuştur.

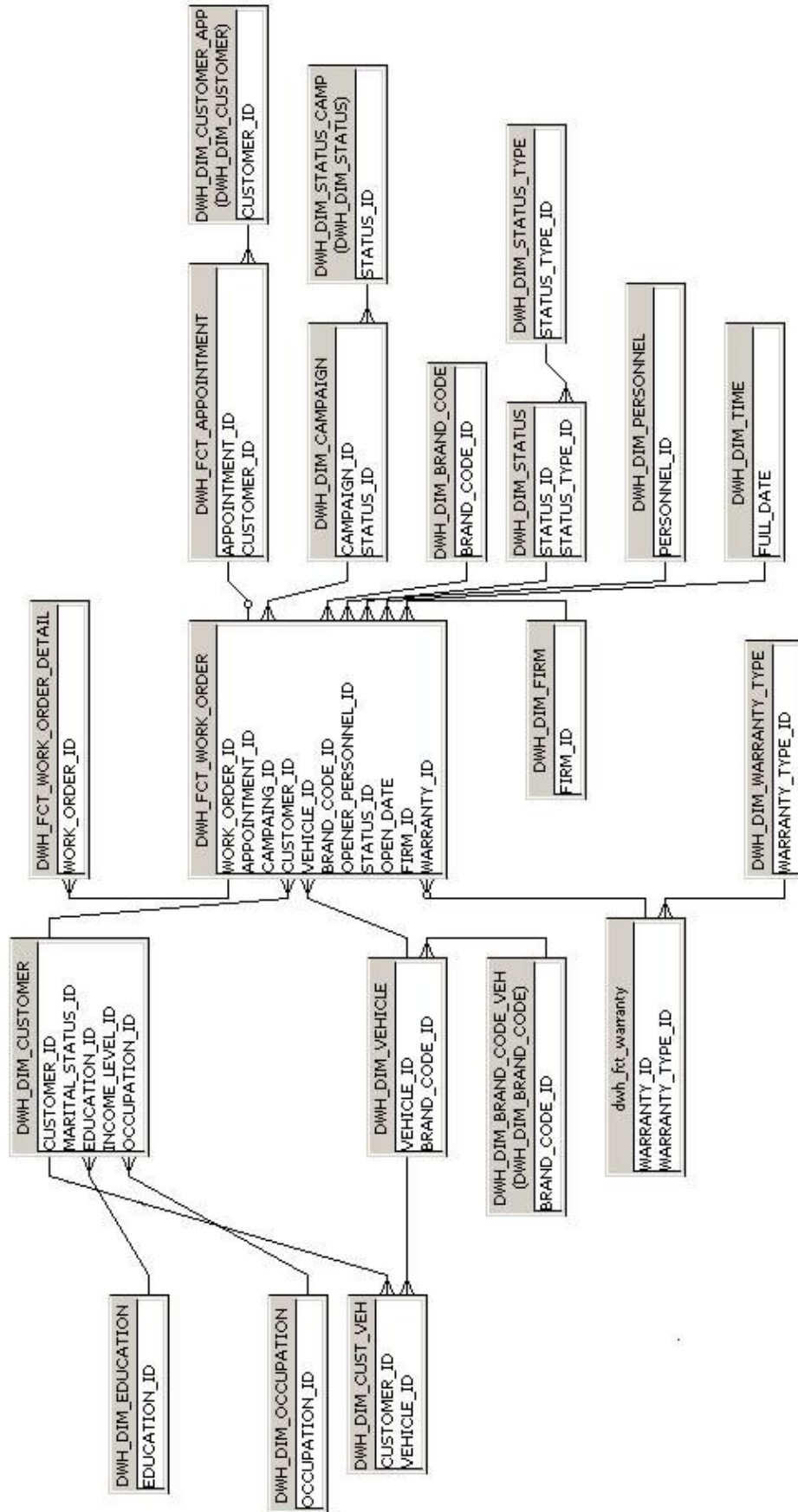
- Boyut Tabloları – İş ile ilgili ilgi parametreleridir. Daha çok Analiz ve Raporlamada verileri filtreleme ve gruplamak için kullanılır. Operasyonel Sistemlerdeki referans tabloları benzerler ve ilişkili özellikler içerebilirler.
- Olgu Tabloları – Bir organizasyonun izlerini ölçen tablolardır. Bir olgu tablosu bir grup boyut tablosu tarafından tanımlanmıştır. Olgu Tabloları tipik olarak operasyonel sistemler tarafından üretilirler.

Veri Ambarı tasarımında bağlantı (join) sayısını azaltmak için denormalizasyon yoluna gidilmiştir. Örneğin; Operasyonel sistemde'de Müşteri ve Müşteri Detay olarak bulunan iki ayrı tablo tek bir tablo adı altında toplanmıştır.

Bu işlem:

2. Boyut olarak Veri tabanını artırmıştır.
3. Kayıt ekleme süresini artırmıştır.

Pek de karlı gibi gözükmeyen bu işlem aslında denormalizasyonun ilk kuralıdır. Veri Ambarlarında amaç veri tabanı boyutunu ve kayıt ekleme süresini azaltmak değil sorguların en kısa sürede görüntülenmesidir. Şekil 10. deki gibi bağlantı sayısını azalttığımız için sorgu operasyonel sistemde çalıştırılmasına göre çok daha çabuk yanıt verecektir.



Şekil 10: DOAS SSH Veri Ambarı Kavramsal Model Tasarımı

### 3.1.1. Boyut Tabloları

Boyut tabloları oluşturulurken her tablonun başına DWH\_DIM\_ yani veri ambarı ve Dimension (Veri Ambarı Boyut Tablosu) anlamına gelen kısaltmalar eklenmiştir. Veri Tabanı yöneticisi ve kullanıcıların tablolara ilk bakışta ne tarz bir tablo olduğunu kolayca anlayabilmeleri için bu yönteme kullanılmıştır.

#### 3.1.1.1. Marka Kodu

Tablo Adı: DWH\_DIM\_BRAND\_CODE

Table - dbo.D...IM_BRAND_CODE			
	Column Name	Data Type	Allow Nulls
🔑	BRAND_CODE_ID	varchar(2)	<input type="checkbox"/>
	BRAND_CODE_DESC	varchar(50)	<input type="checkbox"/>

Şekil 11: Marka Kodu

Marka Kodu tablosu Doğu Otomotiv bünyesinde satılan her marka aracın veri tabanında temsil edildiği yerdir. Her bir Araç Marka Tanımına karşılık bir Marka Kodu tutuluyor, farklı tablolardan Araç Marka bilgilerine ulaşırken “Marka Kodu” kullanılarak çağırılıyor.

BRAND\_CODE\_ID, Marka tablosunda her Marka’ya ait önceden tanımlanmış anahtar alandır. Örneğin AUDI Markası için “A”, PORSCHE için “P”, SEAT için ise “S” anahtar değerleri tablo üzerinde tutulmaktadır.

#### 3.1.1.2. Kampanya

Tablo Adı: DWH\_DIM\_CAMPAIN

Table - dbo.D...DIM_CAMPAIN			
	Column Name	Data Type	Allow Nulls
	CAMPAIN_ID	int	<input type="checkbox"/>
	CAMPAIN_DESC	varchar(50)	<input checked="" type="checkbox"/>
	CAMPAIN_START_D...	datetime	<input checked="" type="checkbox"/>
	CAMPAIN_END_TIME	datetime	<input checked="" type="checkbox"/>
	STATUS_ID	smallint	<input checked="" type="checkbox"/>

Şekil 12: Kampanya

Kampanya tablosu, Servis Kampanyalarının tutulduğu tablodur. CAMPAIGN\_START\_DATE ve CAMPAIGN\_END\_DATE Kampanya Başlangıç ve Bitiş tarihleri göstermektedir. STATUS\_ID ise STATU Tablosundaki (DWH\_DIM\_STATUS ) STATU\_TANIMINI belirtmektedir.

Örneğin, Volkswagen Yaza Hazırlık Kampanyasını CAMPAIGN\_DESC (Kampanya Tanımı) ve Kampanya Başlangıç ve Bitiş tarihleri (CAMPAIGN\_START\_DATE, CAMPAIGN\_END\_DATE) ise 01.05.2007 24:00:00, 01.09.2007 24:00:00 olarak kaydedilmiştir. Kampanyanın Durumu ile ilgili bilgiyi ise (STATUS\_ID) alanında tutulmaktadır.

Kampanya tablosu kaynak sistemden denormalize edilmiş şekilde veri ambarında da oluşturulmuştur.

### 3.1.1.3. Müşteri Araç İlişki Tablosu

Tablo Adı: DWH\_DIM\_CUST\_VEH

Table - dbo.D... DIM_CUST_VEH			
	Column Name	Data Type	Allow Nulls
🔑	CUST_VEH_ID	int	<input type="checkbox"/>
	CUSTOMER_ID	int	<input checked="" type="checkbox"/>
	VEHICLE_ID	int	<input checked="" type="checkbox"/>
	STATUS_ID	smallint	<input checked="" type="checkbox"/>
	STATUS_TYPE_ID	smallint	<input checked="" type="checkbox"/>

Şekil 13: Müşteri Araç İlişki Tablosu

Müşteri Araç İlişki Tablosun da CUST\_VEH\_ID Veri Ambarı tasarımın da IDENTITY alan olarak tutulmuştur. CUSTOMER\_ID, Müşteri Tablosundaki CUSTOMER\_ID ile ilişki FOREIGN KEY (Yabancı Anahtar) dir. VEHICLE\_ID ise Araç Tablosundaki VEHICLE\_ID ile ilişkilendirilmiş FOREIGN KEYdir. STATUS\_ID, bulunduğu kayda ait ilişki durumunu göstermek için tanımlanmıştır. STATUS\_TYPE\_ID, İlişki tipini belirtmektedir. STATUS\_ID ve STATUS\_TYPE\_ID alanları STATUS ve STATUS\_TYPE tablolarında STATUS ve STATUS\_TYPE\_ID alanları ile ilişkilendirilmiştir.

Örneğin, CUSTOMER\_ID değeri 8004399 olan bir AHMET TURGUT adlı Müşterinin VEHICLE\_ID değeri 310929 (METALİK SİYAH PORSCHE CAYENNE) Aracına ait STATUS\_ID değeri 2 (Araç Sahibi) ve STATUS\_TYPE\_ID değeri 1 (Araç Sahibiyet Durum göstergesine ait bir kayıt) eklenmiştir.

#### 3.1.1.4. Müşteri

Tablo Adı: DWH\_DIM\_CUSTOMER

Table - dbo.D..._DIM_CUSTOMER			
	Column Name	Data Type	Allow Nulls
🔑	CUSTOMER_ID	int	<input type="checkbox"/>
	FIRST_NAME	varchar(50)	<input type="checkbox"/>
	MIDDLE_NAME	varchar(50)	<input checked="" type="checkbox"/>
	LAST_NAME	varchar(50)	<input type="checkbox"/>
	GENDER	varchar(1)	<input checked="" type="checkbox"/>
	BIRTH_DATE	datetime	<input checked="" type="checkbox"/>
	BIRTH_DATE_INT	int	<input checked="" type="checkbox"/>
	PHONE_HOME	varchar(50)	<input checked="" type="checkbox"/>
	PHONE_WORK	varchar(50)	<input checked="" type="checkbox"/>
	PHONE_MOBILE	varchar(50)	<input checked="" type="checkbox"/>
	FAX	varchar(50)	<input checked="" type="checkbox"/>
	EMAIL	varchar(50)	<input checked="" type="checkbox"/>
	PASSPORT_NO	varchar(50)	<input checked="" type="checkbox"/>
	MARITAL_STATUS_ID	smallint	<input checked="" type="checkbox"/>
	EDUCATION_ID	smallint	<input checked="" type="checkbox"/>
	INCOME_LEVEL_ID	smallint	<input checked="" type="checkbox"/>
	OCCUPATION_ID	smallint	<input checked="" type="checkbox"/>
	CREATE_PERSONNEL_ID	int	<input type="checkbox"/>
	CREATE_DATE	datetime	<input type="checkbox"/>
	CREATE_DATE_INT	int	<input type="checkbox"/>
	FATHER_NAME	varchar(50)	<input checked="" type="checkbox"/>
	MOTHER_NAME	varchar(50)	<input checked="" type="checkbox"/>
	NATIONALITY	varchar(50)	<input checked="" type="checkbox"/>
	TAX_NUMBER	varchar(50)	<input checked="" type="checkbox"/>
	STATUS_ID	smallint	<input type="checkbox"/>
	STATUS_TYPE_ID	smallint	<input type="checkbox"/>
	CUSTOMER_TYPE	varchar(50)	<input type="checkbox"/>
	CUST_SEQ_NUM	int	<input checked="" type="checkbox"/>

Şekil 14: Müşteri Tablosu

Müşteri Tablosu Kaynak Sistemden denormalize edilerek veri ambarı sisteminde tanımlanmıştır. OLTP Sistemde Customer ve Customer\_Detail adında iki tane tablo bulunuyordu. Customer\_Detail tablosu Müşterilere ait Demografik

bilgilerin tutulduğu tabloydu, veri ambarı tasarımında Customer\_Detail tablosundaki Demografik bilgilerin bulunduğu alanlar Müşteri Tablosuna eklenerek denormalizasyon işlemi gerçekleştirilmiştir. Burada bağlantılardan kurtulup istenen verilere tek bir sorguda erişilebilmesi sağlanmıştır. CUSTOMER\_ID (IDENTITY) Her Müşteri ye ait olan tekil sayısal bir değerdir. FIRST\_NAME, MIDDLE\_NAME, LAST\_NAME, GENDER, BIRTH\_DATE, BIRTH\_DATE\_INT, PHONE\_HOME, PHONE\_WORK, PHONE\_MOBILE, FAX, EMAIL, PASSPORT\_NO, FATHER\_NAME, MOTHER\_NAME, NATIONALITY, TAX\_NUMBER Müşteriye tablosunda tutulan kişisel özlük bilgileridir. MARITAL\_STATUS\_ID (yabancı anahtar) değeri Evlilik Durumunu gösterir ve MARITAL\_STATUS Tablosunda MARITAL\_STATUS\_ID alanı ile ilişkilendirilmiştir. EDUCATION\_ID (yabancı anahtar), değeri Müşteri ait Eğitim Seviyesi Bilgisini göstermektedir. EDUCATION\_ID alanı EDUCATION Tablosunda EDUCATION\_ID alanı ile ilişkilendirilmiştir. OCCUPATION\_ID (yabancı anahtar) değeri Müşterinin Hangi Meslek Grubuna ait olduğu bilgisini tutmaktadır. OCCUPATION\_ID alanı OCCUPATION Tablosunda OCCUPATION\_ID alanı ilişkilendirilmiştir. CREATE\_PERSONEL\_ID, CREATE\_DATE ve CREATE\_DATE\_INT alanları Kaydın hangi Personel tarafından, hangi tarihte eklendiğini göstermektedir. CREATE\_PERSONEL\_ID (FOREIGN\_KEY) alanı PERSONEL tablosunda CREATE\_PERSONEL\_ID ile ilişkilendirilmiştir. STATUS\_TYPE\_ID, İlişki tipini belirtmektedir. STATUS\_ID ve STATUS\_TYPE\_ID alanları STATUS ve STATUS\_TYPE tablolarında STATUS ve STATUS\_TYPE\_ID alanları ile ilişkilendirilmiştir. CUSTOMER\_TYPE, Müşteri Tipini “Bireysel” ve “Kurumsal” olarak tutmaktadır. CUST\_SEQ\_NUM Datanın doğruluğunu test etmek için kullanılıyor. Sadece veri tabanı yöneticisi tarafından görüntülenebilir. Raporlama ekranlarında görüntülenmemektedir.

### 3.1.1.5. Eğitim

Table - dbo....DIM_EDUCATION			
	Column Name	Data Type	Allow Nulls
	EDUCATION_ID	smallint	<input type="checkbox"/>
	EDUCATION_DESC	varchar(50)	<input checked="" type="checkbox"/>

Tablo Adı:  
DWH\_DIM\_EDUCATION

Şekil 15: Eğitim

Eğitim Tablosunda,

EDUCATION\_ID, IDENTITY alan olarak tutuluyor. EDUCATION\_DESC ise Eğitim Tanımı bilgilerini içeriyor.

Örneğin; EDUCATION\_DESC alanında tutulan değerler, İlköğretim öncesi, İlköğretim, Lise, Yüksek Okul, Üniversite, Master ve Doktora sayabiliriz.

### 3.1.1.6. Kurum

Tablo Adı: DWH\_DIM\_FIRM

Table - dbo.DWH_DIM_FIRM			
	Column Name	Data Type	Allow Nulls
	FIRM_ID	int	<input type="checkbox"/>
	FIRM_DESC	varchar(50)	<input checked="" type="checkbox"/>
	FIRM_ADRESS	varchar(250)	<input checked="" type="checkbox"/>
	FIRM_PHONE	varchar(50)	<input checked="" type="checkbox"/>
	FIRM_FAX	varchar(50)	<input checked="" type="checkbox"/>
	CONTACT_PERSON	varchar(50)	<input checked="" type="checkbox"/>

Şekil 16: Kurum

Kurul Tablosunda FIRM\_ID (IDENTITY) alandır. FIR\_DESC firma adlarının tutulduğu alandır. FIRM\_ADRESS, FIRM\_PHONE, FIRM\_FAX ve CONTACT\_PERSON alanları verilere daha kolay ve performans açısından daha az bağlantı kullanarak erişebilmek için Kaynak sistemden denormalize edilerek DATA WAREHOUSE sistemde oluşturulan alanlardır.

Örneğin; FIRM\_DESC için Doğu Otomotiv Servis Ticaret A.Ş. de Servis Hizmeti verilen Bayiler olarak Doğu Oto Maslak, Doğu Oto Ankara, Doğu Oto Kartal gibi bölgeleri sayabiliriz.

### 3.1.1.7. Personel

Tablo Adı : DWH\_DIM\_PERSONNEL



Table - dbo.D..._DIM_PERSONNEL			
	Column Name	Data Type	Allow Nulls
🔑	PERSONNEL_ID	int	<input type="checkbox"/>
	PERSONNEL_NAME	varchar(50)	<input checked="" type="checkbox"/>
	PERSONNEL_SURNAME	varchar(50)	<input checked="" type="checkbox"/>
	PERSONNEL_START_D...	datetime	<input checked="" type="checkbox"/>
	PERSONNEL_END_DATE	datetime	<input checked="" type="checkbox"/>
	PERSONNEL_STATUS_ID	smallint	<input checked="" type="checkbox"/>
	PERSONNEL_GENDER	varchar(50)	<input checked="" type="checkbox"/>
	PERSONNEL_MARITAL...	varchar(50)	<input checked="" type="checkbox"/>
	PERSONNEL_BIRTHDATE	datetime	<input checked="" type="checkbox"/>

Şekil 17: Personel

Personel tablosunda, PERSONNEL\_ID IDENTITY alanıdır. PERSONEL\_NAME, PERSONEL\_SURNAME, PERSONNEL\_START\_DATE, PERSONNEL\_END\_DATE, PERSONNEL\_GENDER, PERSONEL\_MARITAL\_STATUS VE PERSONEL\_BIRTHDATE Personele Ait Özlük bilgileridir. Bu alanlar Kaynak Sistemden denormalize edilerek veri ambarı tasarımına aktarılmıştır. PERSONEL\_STATUS\_ID alanı ise yabancı anahtardır. PERSONEL\_STATUS\_ID alanı PERSONNEL tablosunda PERSONNEL\_STATUS\_ID alanı ile ilişkilendirilmiştir.

### 3.1.1.8. Statü

Tablo Adı : DWH\_DIM\_STATUS

Table - dbo.DWH_DIM_STATUS			
	Column Name	Data Type	Allow Nulls
🔑	STATUS_ID	smallint	<input type="checkbox"/>
	STATUS_TYPE_ID	smallint	<input type="checkbox"/>
	STATUS_DESC	varchar(50)	<input type="checkbox"/>

Şekil 18: Statü

Statü tablosunda STATUS\_ID alanı IDENTITY alanıdır. STATUS\_TYPE\_ID alanı yabancı anahtar alanıdır ve STATUS\_TYPE tablosundaki STATUS\_TYPE\_ID alanı ile ilişkilendirilmiştir. STATUS\_DESC alanı tabloda tutulan statü durumu ile ilgili bilgileri içermektedir.

Örneğin; STATUS\_ID değeri 1 STATUS\_TYPE\_ID değeri 1 ve STATUS\_DESC alanı Araç Sahibi bilgisini tutmaktadır. Bu bilgiye Müşteri Araç

İlişki tablosundaki STATUS\_ID alanından erişmek mümkündür. Boyut tabloların birçoğundan STATUS Tablosuna ilişki mevcuttur. Sorgu performansını artırmak için STATUS Tablosunda STATUS\_ID alanı üzerinde indeks oluşturulmuştur. Bu şekilde STATUS tablosuna ilişkilendirilmiş her sorguda SQL sunucusunun yanıt süresi indekslenmemiş sorguya göre hızlanmış olacaktır.

### 3.1.1.9. Statü Tipi

Tablo Adı : DWH\_DIM\_STATUS\_TYPE

Table - dbo.D...IM_STATUS_TYPE			
	Column Name	Data Type	Allow Nulls
🔑	STATUS_TYPE_ID	smallint	<input type="checkbox"/>
	STATUS_TYPE_DESC	varchar(50)	<input type="checkbox"/>

Şekil 19: Statü Tipi

STATUS\_TYPE tablosunda STATUS\_TYPE\_ID alanı IDENTITY alanıdır. STATUS\_TYPE\_DESC alanı ise Statü Tipi bilgisini barındırmaktadır.

Boyut tabloların birçoğundan STATUS\_TYPE Tablosuna ilişki mevcuttur. Sorgu performansını artırmak için STATUS\_TYPE Tablosunda STATUS\_\_TYPE\_ID alanı üzerinde indeks oluşturulmuştur. Bu şekilde STATUS\_TYPE tablosuna ilişkilendirilmiş her sorguda SQL sunucusunun yanıt süresi indekslenmemiş sorguya göre hızlanmış olacaktır.

### 3.1.1.10. Zaman

Tablo Adı: DWH\_DIM\_TIME

Table - dbo.DWH_DIM_TIME			
	Column Name	Data Type	Allow Nulls
🔑	TIME_ID	int	<input type="checkbox"/>
	FULL_DATE	datetime	<input type="checkbox"/>
	MONTH_NUMBER	smallint	<input type="checkbox"/>
	YEAR_NUMBER	smallint	<input type="checkbox"/>
	DAY_NUMBER	smallint	<input type="checkbox"/>
	DAY_NUMBER_IN_MO...	smallint	<input type="checkbox"/>
	DAY_NUMBER_IN_WEEK	smallint	<input type="checkbox"/>
	DAY_NAME	varchar(50)	<input type="checkbox"/>
	MONTH_NAME	varchar(50)	<input type="checkbox"/>
	IS_WEEKEND	smallint	<input type="checkbox"/>
	DATE_INT	int	<input type="checkbox"/>

Şekil 20: Zaman

Zaman tablosunda, TIME\_ID alanı IDENTITY alanıdır. Zaman tablosu KAYNAK sistem de bulunmamaktadır. Zaman tablosunu veri ambarı üzerinde oluşturmamızın asıl nedeni performans kaybını ortadan kaldırmak ve tarih, gün, ay ve yıl parametrelerine daha kolay erişebilmektir. FULL\_DATE alanında belli bir tarih başlangıç değerinden belli bir bitiş değerine kadar tüm günler tutulmaktadır. SQL Sorguları çalıştırıldığında belli bir tarihteki Gün (DAY\_NUMBER), Ay (MONTH\_NUMBER), Yıl (YEAR\_NUMBER) değerlerine ve o tarihteki aydaki gün sayısı (DAY\_NUMBER\_IN\_MONTH), haftaya ait gün sayısı, (DAY\_NUMBER\_IN\_WEEK) günün Türkçe adı, (DAY\_NAME) “Pazartesi, Salı, Çarşamba, .. vb” gibi ayın Türkçe adı (MONTH\_NAME); “Ocak, Şubat, Mart, .. vb” gibi, hafta sonu olup olmadığı (IS\_WEEKEND) ve rakamsal değeri (DATE\_INT), diğer tarihlerle karşılaştırma yapabilmek için tutulmaktadır.

Örneğin FULL\_DATE alanında tutulan 01.08.2007 değeri için MONTH\_NUMBER değeri 8 dir. Normal koşullarda bulunduğumuz aya erişebilmek için SQL sunucusu üzerinde mevcut MONTH fonksiyonunu kullanmamız gerekiyor. Fakat SQL Sorgularında fonksiyon kullanmak çoğu zaman performans açısından olumsuz değerler yaratıyor. Biz bunun yerine bulunduğumuz tarihi ZAMAN Tablomuzdan çekiyoruz ve SQL SERVER 2000’de kullanılan MONTH fonksiyonu yerine ZAMAN tablosundaki MONTH\_NUMBER değerini kullanabiliyoruz. FULL\_DATE alanına performans açısından indeks tanımlanmıştır.

### **3.1.1.11. Araç**

Tablo Adı: DWH\_DIM\_VEHICLE

Araç Tablosunda VEHICLE\_ID alanı IDENTITY olarak tanımlanmıştır. BRAND\_CODE\_ID alanı ise yabancı anahtardır. BRAND\_CODE\_ID alanı Marka Tablosundaki BRAND\_CODE\_ID alanı ile ilişkilendirilmiştir. Tabloda tutulan her Araç için hangi Markaya ait olduğunu belirtmektedir.

Seq\_num Datanın doğruluğunu test etmek için kullanılıyor. Sadece veritabanı yöneticisi tarafından görüntülenebilir. Raporlama ekranlarında görüntülenmemektedir.

Table - dbo.DWH_DIM_VEHICLE			
	Column Name	Data Type	Allow Nulls
?	VEHICLE_ID	int	<input type="checkbox"/>
	BRAND_CODE_ID	varchar(2)	<input type="checkbox"/>
	MODEL_ID	int	<input checked="" type="checkbox"/>
	MODEL_YEAR	varchar(4)	<input checked="" type="checkbox"/>
	CHASSIS	varchar(50)	<input checked="" type="checkbox"/>
	LICENSE_PLATE	varchar(50)	<input type="checkbox"/>
	ENGINE_NUMBER	varchar(50)	<input checked="" type="checkbox"/>
	TRAFFIC_DATE	datetime	<input checked="" type="checkbox"/>
	TRAFFIC_DATE_INT	int	<input checked="" type="checkbox"/>
	COLOR	varchar(50)	<input checked="" type="checkbox"/>
	VEHICLE_DEFINITION	varchar(250)	<input checked="" type="checkbox"/>
	seq_num	int	<input checked="" type="checkbox"/>

Şekil 21: Araç Tablosu

MODEL\_YEAR Aracını model hangi model yılına ait olduğunu, CHASSIS Aracın motor şaşı numarasını, LICENCE\_PLATE Araç plaka bilgisini, ENGINE\_NUMBER Araç motor numarasını, TRAFFICE\_DATE Araç trafiğe çıkış tarihini, COLOR, VEHICLE DEFINITION Araç ait uzun tanım bilgisi tutulan özlük alanlarıdır. TRAFFIC\_DATE\_INT alanı Aracın trafiğe çıkış tarihinin sayısal olarak tutulmuş sürümüdür.

Araç Tablosu, Kaynak Sistemden denormalize edilerek DATA WAREHOUSE sistemde tasarlanmıştır.

### 3.1.1.12. Garanti Tipi

Tablo Adı: DWH\_DIM\_WARRANTY\_TYPE

Table - dbo....WARRANTY_TYPE			
	Column Name	Data Type	Allow Nulls
?	WARRANTY_TYPE_ID	int	<input type="checkbox"/>
	WARRANTY_TYPE_DESC	varchar(50)	<input type="checkbox"/>

Şekil 22: Garanti Tipi

Garanti Tipi Tablosu, WARRANTY\_TYPE\_ID alanı IDENTITY olarak tanımlanmıştır. WARRANTY\_TYPE\_DESC alanı Garanti Tanımı bilgisini içermektedir. WARRANTY\_ID İş Emri Tablosunda WARRANTY\_ID ile ilişkilendirilmiştir.

Örneğin; WARRANTY\_TYPE\_DESC alanına “1 Yıllık Servis Bakımı Garantisi” örneği verebiliriz.

### 3.1.2. Olgu Tabloları

Olgu Tabloları oluşturulurken, Tabloların Kaynak Sistemde sorulabilecek sorulara mümkün olduğunca en ayrıntılı bir şekilde cevap verebilecek düzeye getirilmeye çalışılmıştır. Bu ayrıntı düzeyine “Grain” bu düzeyin ölçümlenmesine de “Granularity” denilmektedir.

Uygulamamızdaki Olgu Tabloları Randevu, İş Emri ve İş Emri Detayı tablolarından oluşmaktadır.

Olgu tabloları oluşturulurken her tablonun başına DWH\_FCT\_ yani Datawarehouse Dimension (Veri Ambarı Olgu Tablosu) anlamına gelen ifadeler eklenmiştir. Veri Tabanı yöneticisi ve kullanıcıların tablolara ilk bakışta ne tarz bir tablo olduğu anlamaları için bu kısaltmalar eklenmiştir

#### 3.1.2.1. Randevu Tablosu

Tablo Adı: DWH\_FCT\_APPOINTMENT

Table - dbo.D...CT_APPOINTMENT			
	Column Name	Data Type	Allow Nulls
?	APPOINTMENT_ID	int	<input type="checkbox"/>
	CUSTOMER_ID	int	<input checked="" type="checkbox"/>
	APPOINTMENT_DATE	datetime	<input checked="" type="checkbox"/>
	APPOINTMENT_DATE_INT	int	<input checked="" type="checkbox"/>
	VEHICLE_ID	int	<input checked="" type="checkbox"/>
	LICENSE_PLATE	varchar(50)	<input checked="" type="checkbox"/>
	BRAND_CODE_ID	varchar(2)	<input checked="" type="checkbox"/>
	PERSONNEL_ID	int	<input checked="" type="checkbox"/>
	FIRM_ID	int	<input checked="" type="checkbox"/>
	APPOINTMENT_CREATE_DATE	datetime	<input checked="" type="checkbox"/>
	APPOINTMENT_CREATE_DATE_INT	int	<input checked="" type="checkbox"/>
	STATUS_ID	smallint	<input checked="" type="checkbox"/>

Şekil 23:Randevu Tablosu

Randevu tablosu Kaynak Sistemden alınırken denormalize edilmiştir. Randevu tablosundaki APPOINTMENT\_ID alanı IDENTITY alanıdır.

CUSTOMER\_ID; alanı Müşteri tablosunda ilişki olan yabancı anahtardır. VEHICLE\_ID; ise Araç Tablosunda ilişkili olan yabancı anahtardır. BRAND\_CODE\_ID; Marka tablosunda ilişkili olan yabancı anahtardır. PERSONEL\_ID; Personel tablosunda ilişkili olan yabancı anahtardır. FIRM\_ID, Kurum Tablosunda ilişki olan yabancı anahtardır. STATUS\_ID, STATU tablosunda ilişkilendirilmiş yabancı anahtardır. APPOINTMENT\_DATE; Randevu Tarihi, APPOINTMENT\_DATE\_INT; Randevu Tarihinin sayısal değeri, APPOINTMENT\_CREATE\_DATE; Randevunun Sistem tarafından oluşturulma tarihi, APPOINTMENT\_CREATE\_DATE\_INT; Randevunun Sistem tarafından oluşturulma tarihinin sayısal ifadesidir.

Randevu Tablosunda, Müşterinin Aracını Servise getirmeden önce aldığı Randevunun bilgileri tutulur. Araç Bilgisi, Müşteri Bilgisi, Randevuyu alan Personel ve Kurum Bilgisi ve Randevuya ait Randevu Tarihi ve Oluşturulma tarihleri tutulmaktadır.

### **3.1.2.2. İş Emri Tablosu**

Tablo Adı: DWH\_FCT\_WORK\_ORDER

İş Emri Tablosunda WORK\_ORDER\_ID alanı IDENTITY alanıdır. APPOINTMENT\_ID Randevu tablosunda ilişkilendirilmiştir yabancı anahtardır. Her İş Emri Kaydının bir Randevu numarası yani APPOINTMENT\_ID si olmak zorundadır. CAMPAIGN\_ID alanı Kampanya tablosunda ilişkilendirilmiş yabancı anahtardır. CUSTOMER\_ID alanı Müşteri tablosunda ilişkilendirilmiş yabancı anahtardır. Her İş Emri kaydına ait bir Müşteri tanımlanmış olması gerekir. VEHICLE\_ID alanı Araç tablosunda ilişkilendirilmiş yabancı anahtardır. Her İş Emri Kaydına ait bir Araç Tanımlanmış olması gerekmektedir. BRAND\_CODE\_ID alanı Marka tablosunda ilişkilendirilmiş bir yabancı anahtardır. PERSONEL\_ID; Personel tablosunda ilişkili olan yabancı anahtardır. FIRM\_ID, Kurum Tablosunda ilişki olan yabancı anahtardır. STATUS\_ID, STATU tablosunda ilişkilendirilmiş yabancı anahtardır. WARRANTY\_ID, Garanti tablosunda ilişkilendirilmiş yabancı anahtardır.

Table - dbo.D...CT_WORK_ORDER			
	Column Name	Data Type	Allow Nulls
🔑	WORK_ORDER_ID	int	<input type="checkbox"/>
	APPOINTMENT_ID	int	<input checked="" type="checkbox"/>
	CAMPAING_ID	int	<input checked="" type="checkbox"/>
	CUSTOMER_ID	int	<input checked="" type="checkbox"/>
	PAYMENT_TYPE	int	<input checked="" type="checkbox"/>
	LICENSE_PLATE	varchar(50)	<input checked="" type="checkbox"/>
	VEHICLE_ID	int	<input checked="" type="checkbox"/>
	BRAND_CODE_ID	varchar(2)	<input checked="" type="checkbox"/>
	OPENER_PERSONNEL_ID	int	<input checked="" type="checkbox"/>
	MILEAGE	int	<input checked="" type="checkbox"/>
	STATUS_ID	smallint	<input checked="" type="checkbox"/>
	STATUS_TYPE_ID	smallint	<input checked="" type="checkbox"/>
	OPEN_DATE	datetime	<input checked="" type="checkbox"/>
	OPEN_DATE_INT	int	<input checked="" type="checkbox"/>
	CLOSE_DATE	datetime	<input checked="" type="checkbox"/>
	CLOSE_DATE_INT	int	<input checked="" type="checkbox"/>
	FIRM_ID	int	<input checked="" type="checkbox"/>
	FUEL_STATUS_ID	int	<input checked="" type="checkbox"/>
	EXPECTED_AMOUNT	int	<input checked="" type="checkbox"/>
	AMOUNT	int	<input checked="" type="checkbox"/>
	PAID_AMOUNT	int	<input checked="" type="checkbox"/>
	RESPONSIBLE_PERSON...	int	<input checked="" type="checkbox"/>
	CANCEL_EXPLANATION	varchar(250)	<input checked="" type="checkbox"/>
	WARRANTY_ID	int	<input checked="" type="checkbox"/>
	seq_num	int	<input checked="" type="checkbox"/>

**Şekil 24: İş Emri Tablosu**

PAYMENT\_TYPE, İş Emrine ait ödeme bilgisini içermektedir. LICENCE\_PLATE, Araca ait Plaka bilgisini içermektedir. Araca ait Plaka bilgisine VEHICLE\_ID ile Araç Tablosundan erişebilmekte mümkün fakat bu arada ayrıca tutulmasının sebebi, sorgu sırasında doğrudan Araç Plakasından Araç Bilgilerine erişilemeyi mümkün kılmaktır. LICENCE\_PLATE alanı Kaynak sistemde olmamasına rağmen denormalize edilerek İş Emri Tablosuna eklenmiştir. OPENER\_PERSONNEL\_ID, İş Emri kaydını açan Personeline ait PERSONEL\_ID değeridir, PERSONEL tablosuna ilişkilendirilmiştir. MILEAGE, Aracın İş Emri açıldığındaki kilometre değeridir. OPEN\_DATE, İş Emri kaydının açıldığı tarihi içerir, OPEN\_DATE\_INT; İş Emri kaydının açıldığı tarihin sayısal değerinin tutmaktadır. CLOSE\_DATE; İş Emri kaydının kapandığına dair bilgiyi tutmaktadır. CLOSE\_DATE\_INT, İş Emri kaydının kapandığına dair bilgiyi sayısal olarak

tutmaktadır. FUEL\_STATUS\_ID, İş Emri kaydı açıldığında aracın benzin durumunu gösteren bilgi içermektedir. EXPECTED\_AMOUNT, İş Emri Kaydında yapılacak işlerin bedelidir. AMOUNT; Kampanya veya Garanti durumlarında İş Emrinde yapılan işin bedeline yansıyan tutardır. PAID\_AMOUNT; İş Emri kapandıktan sonra Müşteriden tahsil edilen tutar bilgisini içermektedir. RESPONSIBLE\_PERSON, İş Emri Kaydını açan ve kaydı yürüten Personelin bilgisini içermektedir. PERSONEL tablosuyla ilişkilendirilmiştir. CANCEL\_EXPLANATION; İş Emrinin iptal edilmesin durumunda konuyla ilgili iptal açıklamasının tutulduğu alandır.

### 3.1.2.3. İş Emri Detayı Tablosu

Tablo Adı: DWH\_FCT\_WORK\_ORDER\_DETAIL

Table - dbo.D...K_ORDER_DETAIL			
	Column Name	Data Type	Allow Nulls
?	WORK_ORDER_DETAIL_ID	int	<input type="checkbox"/>
	WORK_ORDER_ID	int	<input type="checkbox"/>
	DESCRIPTION	varchar(250)	<input checked="" type="checkbox"/>
	PERSONNEL_ID	int	<input checked="" type="checkbox"/>

Şekil 25: İş Emri Detayı Tablosu

İş Emri Detay Tablosunda, WORK\_ORDER\_DETAIL\_ID alanı IDENTITY alandır. WORK\_ORDER\_ID alanı İş Emri tablosuyla ilişkilendirilmiş yabancı anahtardır. İş Emrinde yapılan her işin detayı bu tabloda tutulur. DESCRIPTION alanında İş Emrinde yapılan her türlü işin tanımı vardır. Örneğin; 30.000 bakımı, Lastik değişimi, Balata, Disk Fren değişimi gibi. PERSONEL\_ID alanı PERSONEL tablosuyla ilişkili yabancı anahtardır.

## 3.2. Veri Akışı

Doğuş Otomotiv Servis Ticaret A.Ş. Servis Noktalarında her firmanın kendi Veri tabanı bulunmaktadır. Her gün sonunda saat 00:00 te firmalar da çalışan bir Job aracılığı ile Veri Tabanında bulunan bilgiler formatlı bir şekilde metin dosyalarına yazdırılır. Bu metin dosyalarında her kolon bir tab karakteri ile ayrıştırılmıştır ve Veri Ambarı Projesinin standartlarına uygun olarak isimlendirilir. Daha önceden kararlaştırılmış formatımıza göre



Firmalsmi\_GünAyYıl.txt olarak kaydedilir. Son olarak bu dosyalar Veri Aktarımının yapılacağı sunucuya ftp teknolojisi kullanılarak transfer edilir.

Veri Aktarımı işlemi gerekli dosyaların varlığının kontrolünden sonra saat 04:00'te başlar. Hazırlanmış olduğumuz ETL işlemi için dosyalarını bu konumdan hazırladığımız düz dosya (flatfile) formatlarını kullanarak okumaktadır ve safhalama (staging) tablolarına aktarmaktadır.

Bir sonraki aşamada Dönüşüm (Transformation) başlamaktadır. DWH tablolarıyla karşılaştırılan veriler güncelleme ve ekleme (Update ve Insert) gerektiğinde yapılmaktadır. eklenen kayıtlara anahtar oluşturma (key\_generation) metodu kullanılarak yeni bir datawarehouse\_ID verilmektedir. (örneğin; work\_order\_id, vehicle\_id)

Son aşama olan Yükleme (Load) aşamasında güncelleme ve ekleme işlemleri dwh tablolarına yansıtılır. Tüm boyut tabloları tek bir metin dosyası üzerinden okunmaktadır.

### **3.2.1. ODBC Veri Kaynağının Tanımlanması**

“Excel, en çok ODBC veya OLE DB veri kaynaklarından veriye erişebilmektedir. Office 2000; MS SQL Server, SQL Server OLAP Services, MS Access 2000, dBase, MS FoxPro, Oracle, Paradox ve metin dosyalarına erişim sürücülerıyla donatılmıştır.”

Çalışmamızda, uygulamamızın Veri tabanı ile ilişki kurabilmesi için ODBC (ODBC=Open Database Connectivity=Açık Veritabanı Bağlanırlığı) veri kaynağının tanımlanmış olması gerekmektedir. ODBC veri kaynağını tanımlama işlemlerini şu şekilde sıralayabiliriz:

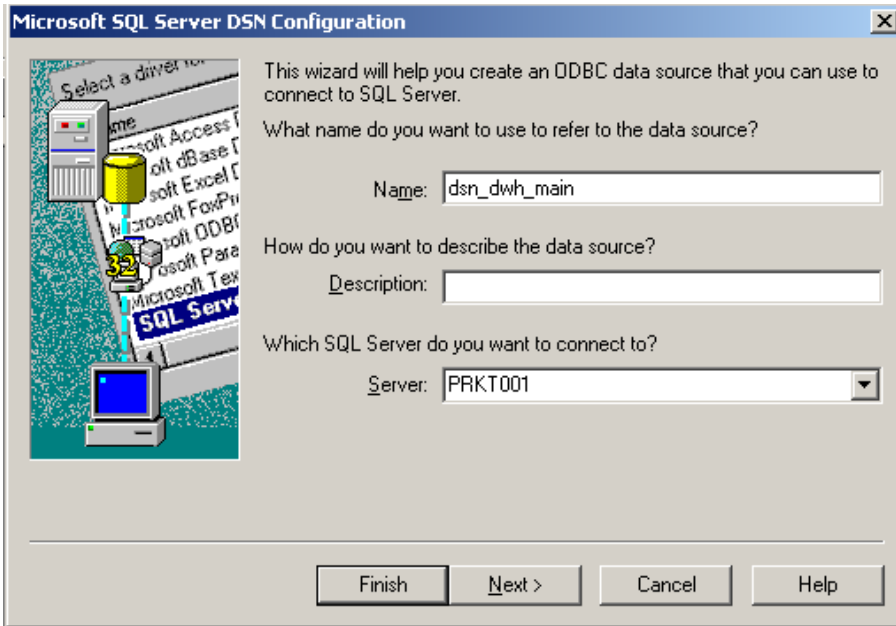
İlk olarak bilgisayarımızın masaüstünde bulunan Bilgisayarım simgesi seçilir. Daha sonra ekrana gelen bilgisayarım simgesi içindeki denetim masası simgesi seçilir ve Şekil-26 de ki denetim masasında yer alan ODBC Veri Kaynakları simgesi üzerinde çift tıklama yapılır.



Şekil 26: Denetim Masasında ODBC Veri Kaynağı Simgesinin Seçimi

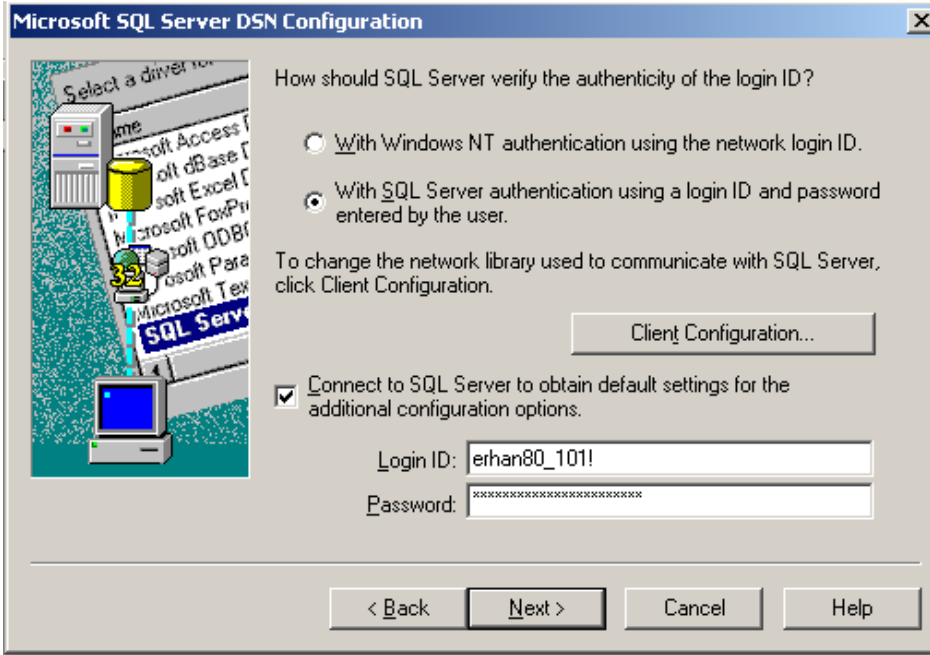
### 3.2.2. Veri Kaynağı Yapılandırılması

ODBC Veri Kaynağı seçildikten sonra Ekle kutucuğuna tıklanarak yeni bir DSN tanımlanır. Veri Tabanı olarak SQL Sunucusu seçilir ve Şekil 27 deki gibi tanımlama yapılır.



Şekil 27: Veri Kaynağı Yapılandırılması

DSN ismi ve Sunucu tanımı yapıldıktan sonra Şekil 28 gibi kullanıcı adı ve şifre tanımı yapılır. İleri kutucuğuna tıklanarak DSN tanımı tamamlanır.



Şekil 28: Veri Kaynağı Yapılandırılması

### 3.2.3. DataIntegrator da Datastore tanımı oluşturmak

Veri aktarımı işleminde Business Objects'in Data Integrator ETL (Extract, Transformation, Load) yazılımı kullanılmaktadır. Piyasada bulunan en güçlü ETL yazılımlarından birisi olan Data Integrator özellikle kullanım kolaylığı ve yüksek performans sebebi ile uygulamamız için seçilmiştir. Veri Aktarımı için yapılması gereken ilk iş Datastoreların tanımlanmasıdır. Datastore, DataIntegrator tarafından kullanılan bağlantı nesnelere dir. Veri Tabanında bulunan tabloların yardımcı veri (metadata) bilgisini repository import etmek için kullanılır. Şeki3.20 de yeni bir Data Store oluşturuyoruz. Bu DataStore Development Veri Tabanına bağlantı için kullanılmaktadır. Development Veri Tabanında ETL sürecinde kullanılan geçici (template) ve safhalama tabloları bulunmaktadır. Şekil 29 da yeni bir Datastore oluşturulurken Datastore adı, tipi, Veri Tabanı tipi, Veri Tabanı Versiyonu, Veri Tabanı Sunucusu adı, Veri Tabanı adı, Kullanıcı adı ve Şifre bilgileri girilmektedir.

Dialog box titled "Create New Datastore" with the following fields and values:

- Datastore name: DS\_DWH\_ADM
- Datastore type: Database
- Database type: Microsoft SQL Server
- Database version: Microsoft SQL Server 2000
- Database server name: PRKT001
- Database name: DWH\_ADMIN
- User name: erhan79\_99!
- Password: \*\*\*\*\*

Buttons: Advanced >>, Show ATL..., OK, Cancel, Apply. An unchecked checkbox labeled "Enable CDC" is also present.

Şekil 29: Datastore tanımı oluşturmak

Şekil 30 Bir Datastore tanımı Veri Ambarı için oluşturulmaktadır. Aynı tanımlar bir önceki Datastore tanımlandığı gibi yapılıyor fakat sadece kullanıcı adı ve şifreleri farklı Veri Tabanlarını gördükleri için ayrıca girilmektedir.

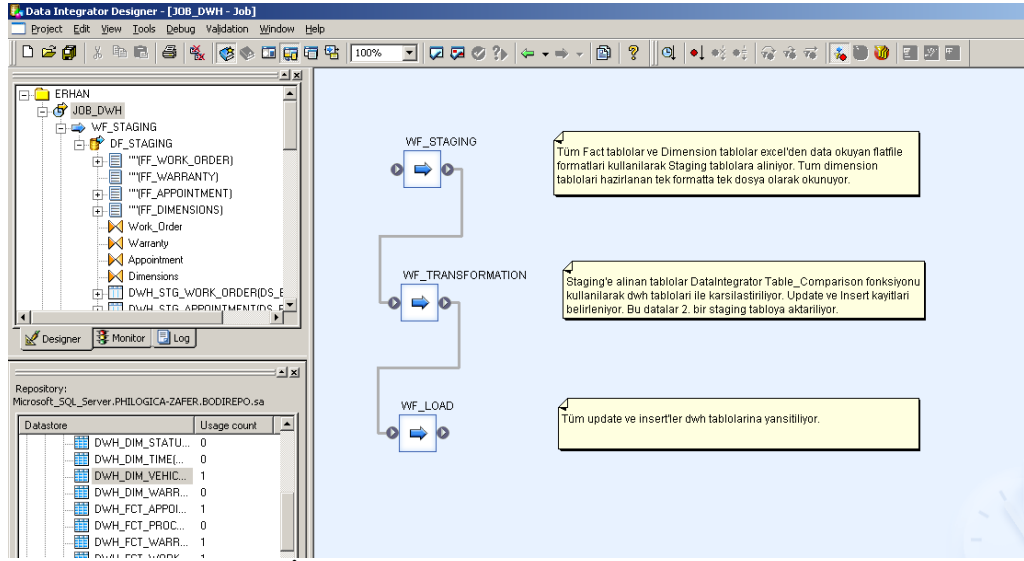
Dialog box titled "Create New Datastore" with the following fields and values:

- Datastore name: DS\_DWH\_MAIN
- Datastore type: Database
- Database type: Microsoft SQL Server
- Database version: Microsoft SQL Server 2000
- Database server name: PRKT001
- Database name: DWH\_MAIN
- User name: erhan80\_101!
- Password: \*\*\*\*\*

Buttons: Advanced >>, Show ATL..., OK, Cancel, Apply. An unchecked checkbox labeled "Enable CDC" is also present.

Şekil 30: Veri ambarı için Datastore tanımı oluşturmak

### 3.2.4. Seçip Çıkarma İşlemi



Şekil 31: Seçip Çıkarma İşlemi

Veri aktarımı üç farklı parçadan oluşmaktadır; Verinin safhalama tablolarına alınması (Extraction), Veri validasyon aşamalarından sonra Veri Ambarı ID'sinin oluşturulması ve güncellenecek ve eklenecek yeni kayıtların belirlenmesi (Transformation) ve son olarak da elde edilen verinin oluşturulan veritabanı tablolarına yüklenmesi (Loading) işlemleridir.

Safhalama aşamasında kaynak olarak metin dosyaları kullanılmıştır. Bu metin dosyalar yazılan bir shell script sayesinde kaynak veri tabanından her gece 02:00 ile 04:00 arasında çekilmektedir. Basit select statement'lerden oluşan bu SQL statementlerinin sonuçları ilgili metin dosyaya yazdırılır ve bu dosyalar DataIntegrator'da oluşturulan DüzDosya Şekil (Flat File Format) nesnesi tarafından okunur. Her DüzDosya Şekil nesnesi "FF\_" karakterleri ile başlar. Bu veri ambarı projesinde kullandığımız standartlardan biridir. Bu DüzDosya Şekil ilgili dosyalara FTP yönetimiyle erişerek veriyi çeker ve safhalama tablolarına paralel olarak yazar. Safhalama tablolarımız "DWH\_STG\_" karakterleri ile başlar. Bu da kullanıcıların hangi tablonun ne amaçla kullanıldığını daha kolay anlaması için belirlediğimiz veri ambarı projesi standartlarından biridir.

Dönüşüm aşamasında safhalama alınan veri önce satır satır geçerlilik kontrolünden geçirilir. Belirlediğimiz veri kalitesi eşğine erişemeyen kayıtlar

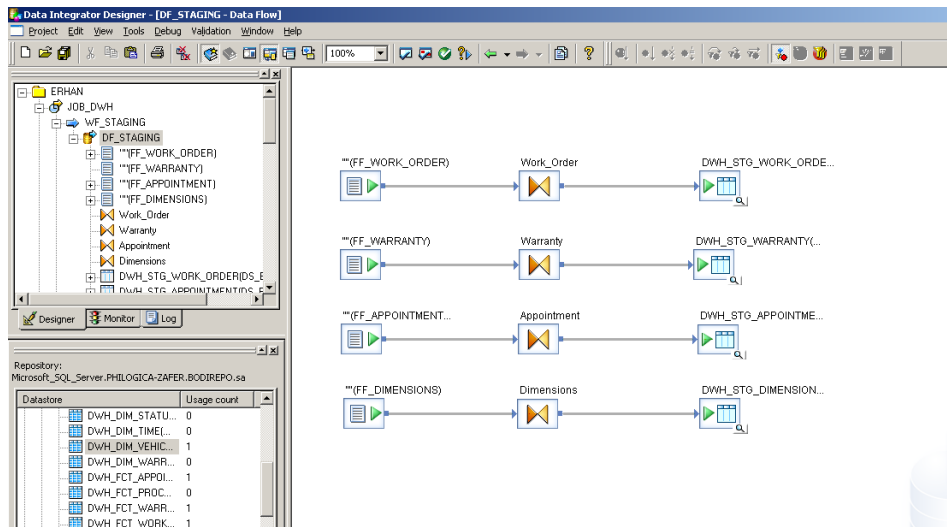
elenerek geri dönüşüm (recycle) tablolarına yüklenir. Bu tablolar IT elemanları tarafından el ile düzeltilirse bir dahaki yüklemde otomatik olarak onlar da veri ambarına yüklenir. Validasyondan geçen her satır veri, tablosuna göre 2 ayrı işlemde geçirilir;

1-Eğer bu bir boyut tablo verisi ise ilgili veri ambarı tablosu ile karşılaştırılır ve eski kayıtlarda herhangi bir değişiklik olup olmadığı kontrol edilir (güncelleme) ve yeni eklenen kayıtlar belirlenir (yeni kayıt ekleme işlemi)

2-.Eğer bu satır bir olgu tablosuna ait ise doğrudan yeni kayıt ekleme işlemi olarak kabul edilir.

Dönüşümün son aşaması olarak her satıra ilgili veri ambarı ID'si atanır, bu satır bir güncelleme ise zaten olan ID veri ambarından sorgulanarak alınır, yeni kayıt ekleme işlemi ise key\_generation metodu kullanılarak yeni bir ID oluşturulur. Bu aşamadaki tablolarımız “DWH\_TMP\_” karakterleri ile başlar, kullanılan tabloların template (geçici) tablo olduğunu gösterir.

Son aşama olarak belirlenen kayıtlar ilgili tablolara paralel olarak yüklenir. Boyut tabloları “DWH\_DIM\_”, Olgu tablolar “DWH\_FCT\_” son olarak Özet tablolar ise “DWH\_SUM\_” karakterleri ile başlar. Özet tablolar Olgu tabloların bir araya getirilmesi (aggregate) ile oluşturulur.



Şekil 32: Seçip Çıkarma İşlemi

### 3.2.5. DüzDosya Örneği

Bu aşamada üç olgu tablosu (Work\_Order, Warranty, Appointment) ve boyut tabloları (tüm boyut değerleri açıkladığımız gibi tek metin dosyadan alınıyor) safhalama tablolarına yükleniyor. Yükleme işlemi paralel olarak gerçekleştiriliyor ve daha önceden belirttiğimiz gibi DüzDosya Şekilleri kullanılıyor.

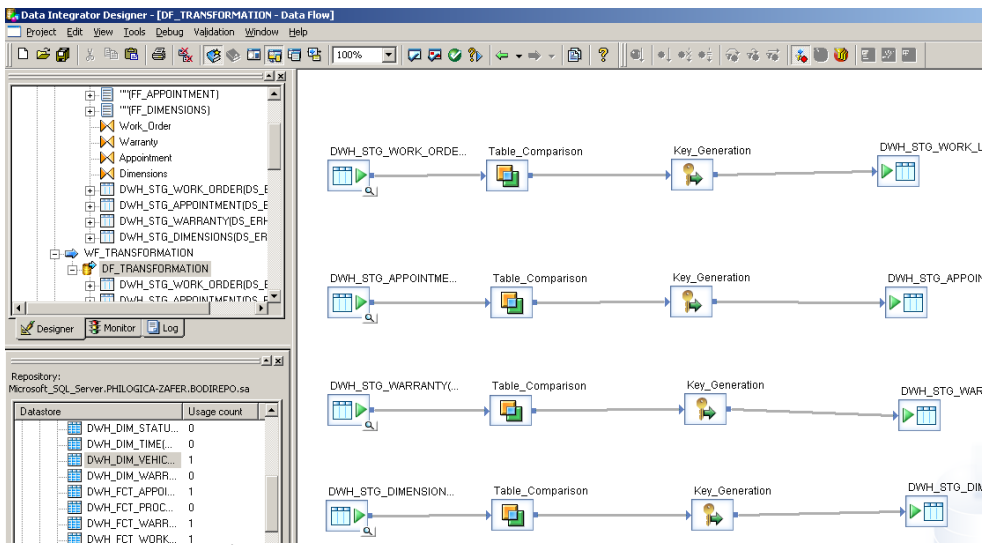
Property	Value
<b>General</b>	
Type	Delimited
Name	FF_WORK_ORDER
Custom transfer program	Yes
Parallel process threads	2
<b>Data File(s)</b>	
Location	Local
Root directory	Confidential
File name(s)	Confidential
<b>Delimiters</b>	
Column	Tab
Row	{new line}
Text	{none}
<b>Default Format</b>	
Escape char	{none}
NULL indicator	{none}
Ignore row marker(s)	{none}
Date	yyyy.mm.dd
Time	hh24:mi:ss
Date-Time	yyyy.mm.dd hh24:mi:ss
<b>Input/Output</b>	
Style	Headers
Skipped rows	0
Skip row header	No
Write row header	No
<b>Custom Transfer</b>	
Program executable	FTP.EXE
User name	ERHAN
Password	*****
Arguments	1,2,4
<b>Locale</b>	
Language	<default>
Code page	<default>
<b>Error handling</b>	

Field Name	Data Type	Field Size
DIMENSION_COD	varchar	100
SOURCE_ID	varchar	100
FIELD_1	varchar	100
FIELD_2	varchar	100
FIELD_3	varchar	100
FIELD_4	varchar	100
FIELD_5	varchar	100
FIELD_6	varchar	100
FIELD_7	varchar	100
FIELD_8	varchar	100
FIELD_9	varchar	100
FIELD_10	varchar	100

SOURCE_ID	FIELD_1	FIELD_2
1		
2		
3		
4		
5		
6		
7		
8		
9		
10		
11		

Şekil 33: DüzDosya Örneği

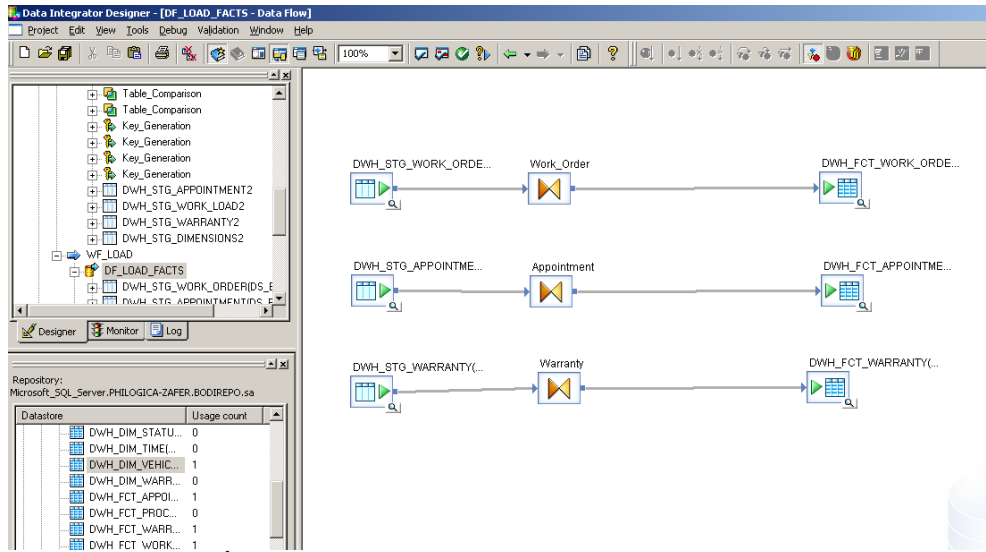
### 3.2.6. Dönüşüm İşlemi



Şekil 34: Dönüşüm İşlemi

Dönüşüm aşamasının gösterildiği bu şekilden de anlaşılacağı gibi safhalama tablolarına alınan veriler Table\_Comparison metodu kullanılarak ilgili tablolara karşılaştırılıp güncellenecek ve yeni eklenecek kayıtlar belirleniyor. Güncellenen kayıtların veri ambarı ID'leri eski kayıttan kontrol edilerek alınıyor, eklenen yeni kayıtlar ise Key\_Generation metodu kullanılarak yeni bir veri ambarı ID'si veriliyor. Sonuç olarak bu kayıtlar ilgili tablolara yazılıyor ve veritabanımıza yapılacak olan doldurma işlemi belirlenmiş oluyor.

### 3.2.7. Yükleme İşlemi



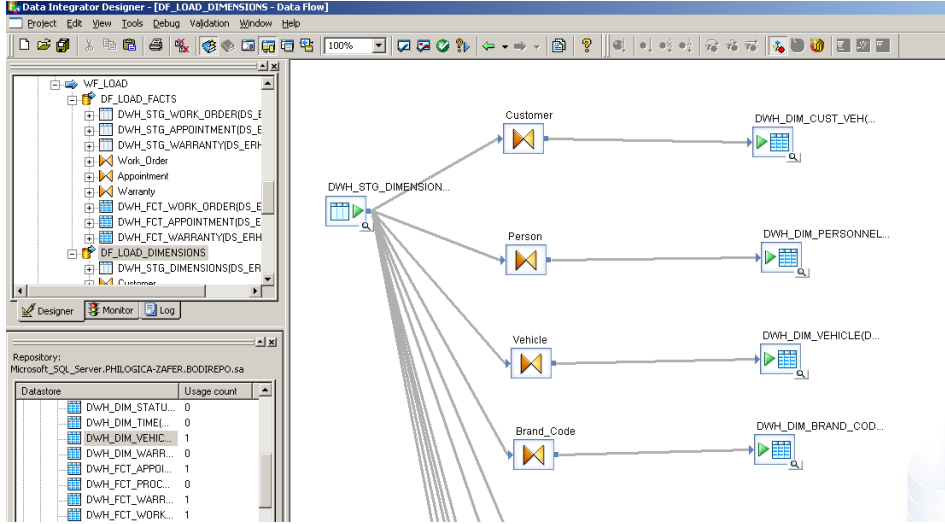
Şekil 35: Yükleme İşlemi

Yükleme aşamasının gösterildiği bu şekilde, veriler belirlenen paralellikte ilgili tablolara yazılıyor ve veri aktarımı işlemi tamamlanıyor.

### 3.2.8. Boyut Tabloların Doldurulması

Boyut tablolarının doldurulmasının gösterildiği bu şekilde, görüldüğü üzere tüm boyut değerleri tek bir tablodan DIMENSION\_CODE\_ID değeri sayesinde ayrıştırılarak okunuyor. Ortada turuncu renkte gördüğünüz ve sorgu olarak adlandırılan dönüşüm nesnesine bir “where” koşulu ekleyerek (DIMENSION\_CODE\_ID =1 gibi) bu ayrıştırma yapılıyor. Tüm boyut değerleri ilgili query'lerce safhalama tablolarından okunuyor





**Şekil 36: Boyut Tablolarının Doldurulması**

### 3.2.9. Universe oluşturmak

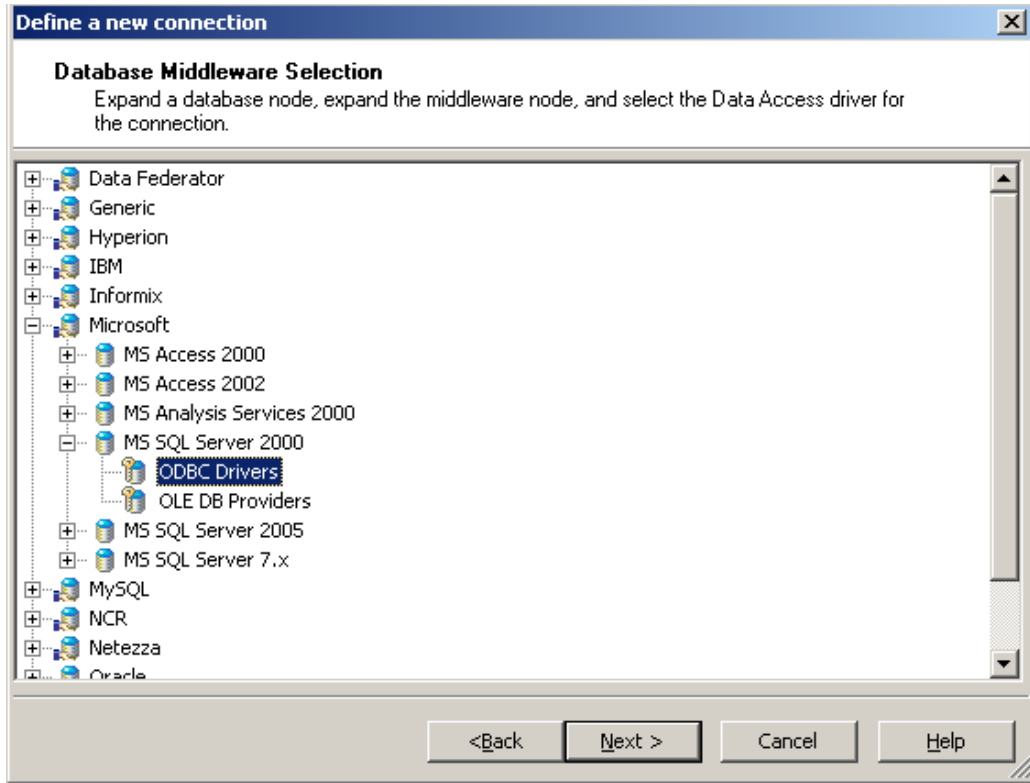
Business Objects Universe, veri tabanı ile rapor oluşturan kullanıcılar arasında bulunan ve sorulan soruları SQL cümleciklerine çeviren ara katmandır. Veri tabanında bulunan ve rapor oluşturan kullanıcılara bir anlam ifade etmeyen kolon isimlerini iş terimlerine çevirir.

Örneğin; DWH\_FCT\_WORK\_ORDER tablosundaki Total\_Amount kolonu bir kullanıcı için çok fazla şey ifade etmeyebilir. Universe aracılığı ile bu kolon İş Emri Tutarı olarak ifade edilebilir. Business Objects Universe sayesinde rapor oluşturan kullanıcıların SQL bilme zorunluluğu da kalmamaktadır.

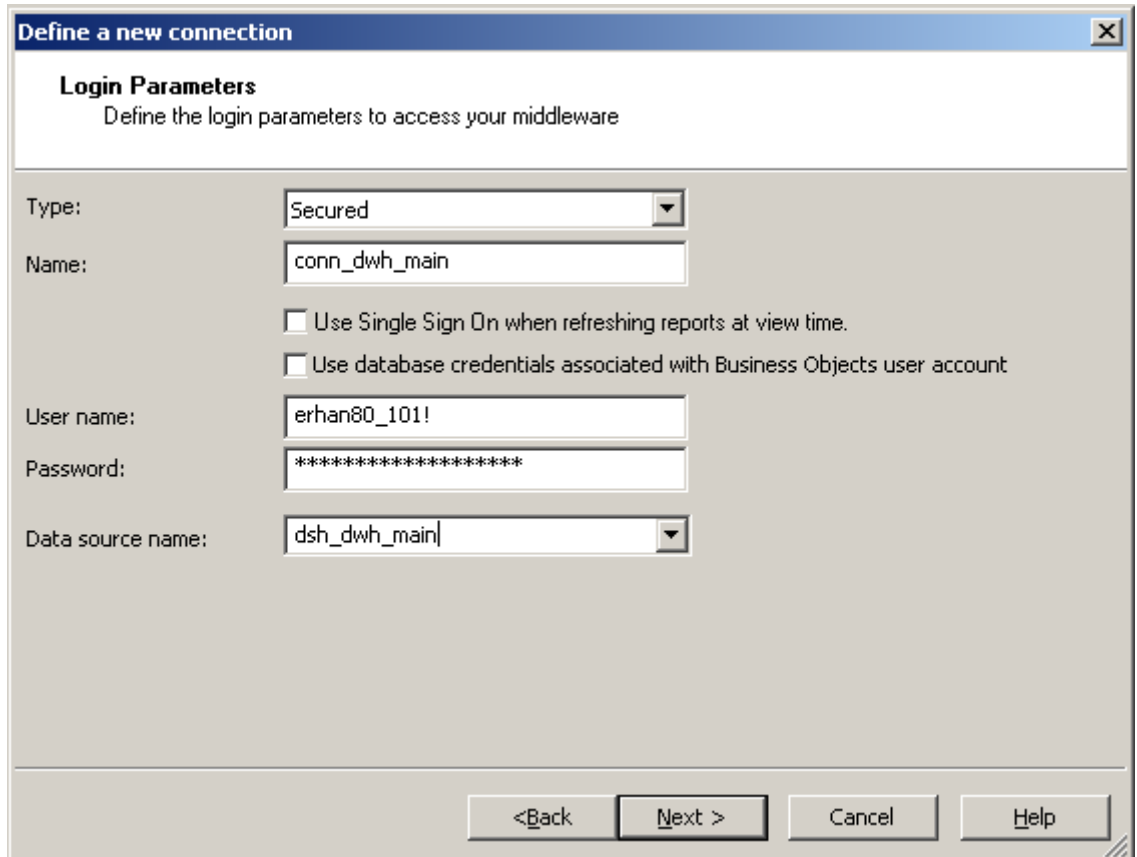
#### 3.2.9.1. Universe Bağlantısı Oluşturmak

Universe oluştururken gerçekleştirilen ilk adım bağlantı tanımlanmasıdır. Şekil 3.28 de görüldüğü gibi ilk adım bağlantının yapılacağı Veritabanı tipinin ve bağlantı teknolojisinin seçilmesidir.

İkinci adım olarak Şekil 38 de görüldüğü gibi kullanıcı adı, şifre ve DSN tanımları yapılır. Bu aşamada önemli olan tip olarak “Secured” bağlantı tipinin seçilmesidir. Secured bağlantı tipi bize oluşturduğumuz Universe’ün sisteme export edilebilmesini ve diğer kullanıcılar tarafından kullanılabilmesini sağlar

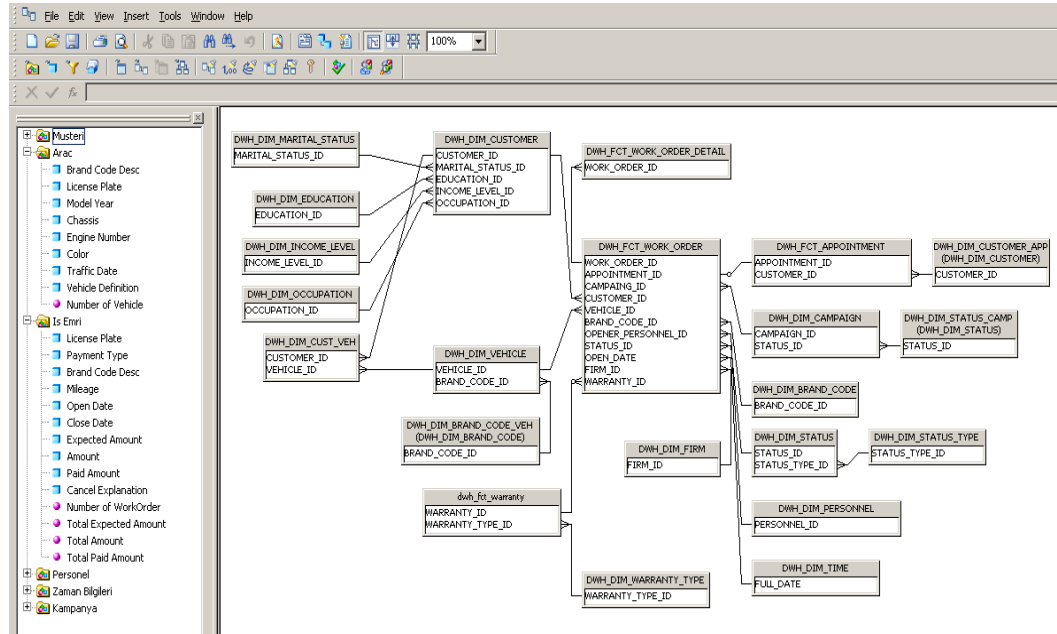


Şekil 37: Universe Bağlantısı Oluşturmak



Şekil 38: Universe Bağlantısı Oluşturmak

Bağlantılar oluşturulduktan sonra bu bağlantının gördüğü Veritabanındaki istenilen tablolar Universe eklenir. İlk iş olarak bu tablolar arasındaki bağlantılar tanımlanır. Bu bağlantılar rapor oluşturulması sırasında doğru SQL in üretilmesi için gerekli olan ilk şarttır. Şekil 39 da görüldüğü gibi tablolar ve bağlantılar sağ bölümde yer almaktadır. Oluşturulacak olan nesnelere ise sol bölümde yer almaktadır. Yeni bir nesne oluşturmak için istenilen kolonun seçilerek sol bölüme taşınması yeterlidir. Sol bölüm farklı klasörler oluşturularak daha düzenli bir yapıya kavuşturulabilir. Rapor oluşturulması sırasında Web Intelligence ekranında kullanıcıların görecekleri nesnelere Universe de sol tarafta bulunan nesnelerdir.



Şekil 39: Universe Oluşturma

### 3.3. Raporlama

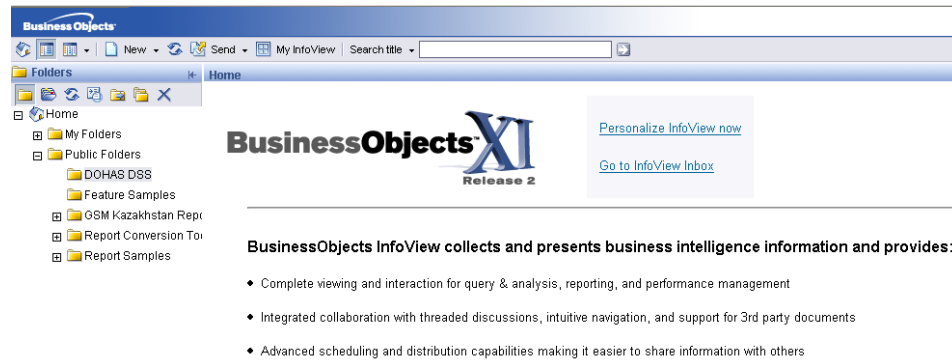
Veri Ambarı Uygulamamızda Rapor oluşturmak için Business Objects XI programı kullanılmaktadır. Business Objects aynı isimli Fransız bir şirketinin sahibi olduğu en yaygın olarak kullanılan raporlama yazılımlarından biridir. Yazılımın genel olarak amacı, firmaların ellerinde bulunan fakat değerlendirilemeyen bilgilerin iş zekâsı raporlarına çevirerek karar destek sistemlerine yardımcı olmasıdır.

Rapor oluşturmak için kullanıcı bilgisayarlarına bir yazılımın yüklenmesi zorunluluğu yoktur, bu da Business Objects'in en güçlü özelliklerinden birisidir. Bir web tarayıcısı aracılığı ile sisteme bağlanılır ve java teknolojisi kullanarak raporlar oluşturulabilir. Rapor oluşturma öncesinde yapılması gereken en önemli işlem, Business Objects Universe'ün oluşturulmasıdır.

Universe kısaca veri tabanında bulunan ve rapor oluşturan kullanıcılara anlamsız gelen tablo kolon isimlerini anlamlı iş terimlerine çeviren ara katmandır. Universe oluşturulması sırasında tıpkı veri tabanı tasarımında olduğu gibi tablolar arasındaki ilişkiler tanımlanır. Böylece seçilen nesnelere göre SQL de otomatik olarak oluşturulur.

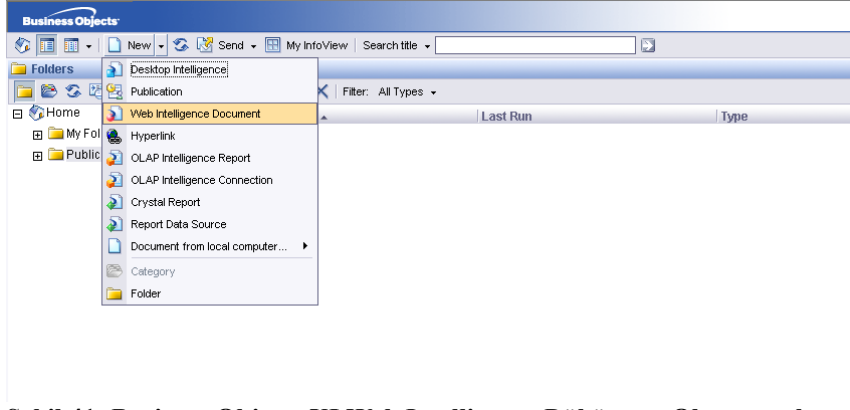
### 3.3.1. Business Objects'te Örnek Rapor Oluşturma

Rapor hazırlamak için Business Objects XI programını çalıştırıyoruz. Sisteme kullanıcı ve şifre bilgilerini girdikten sonra karşımıza aşağıdaki ekran görüntülenir. Burada görülen sol çerçevede bulunan klasörler Business Objects yöneticileri tarafından sistemde oluşturulur. Bizim raporlarımız sistemde DOHAS DSS adlı klasörde bulunmaktadır.



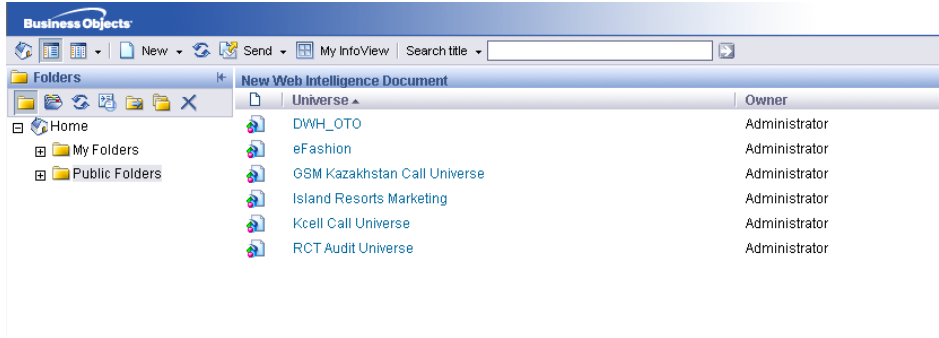
Şekil 40: Business Objects XI Açılış Ekranı

Rapor oluşturmak için yapılması gereken ilk işlem New menüsünden yeni bir “Web Intelligence Document” seçeneğinin seçilmesidir.



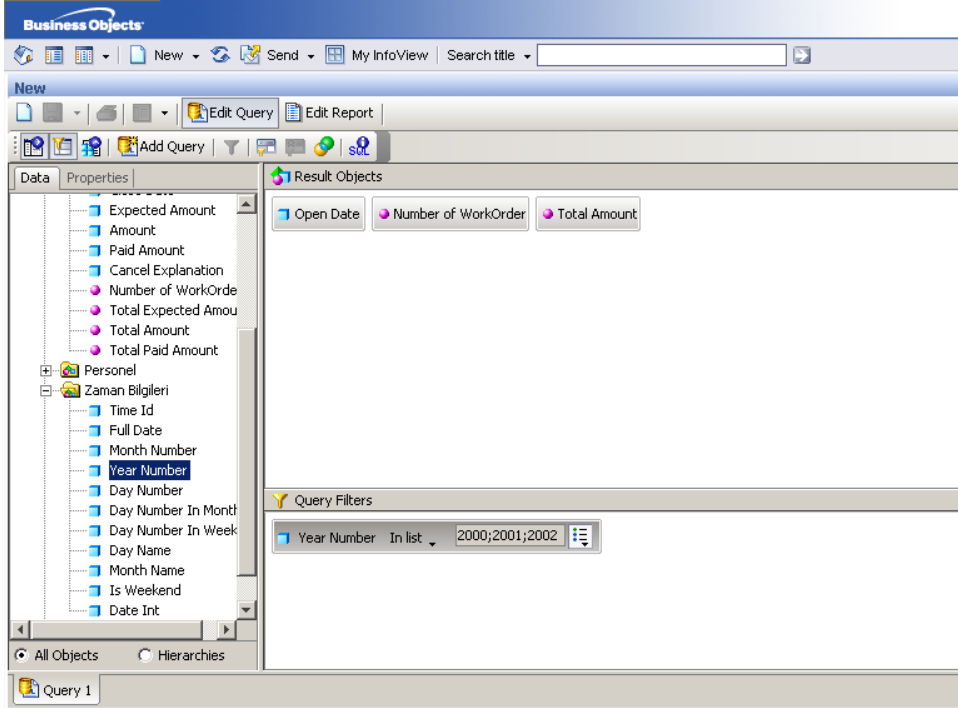
**Şekil 41: Business Objects XI Web Intelligence Dökümanı Oluşturmak**

Yeni “Web Intelligence Document” seçildikten sonra oluşturacağımız rapor için hangi universe’ü kullanacağımızı belirtmemiz gerekmektedir. Belirttiğimiz gibi universe veri tabanı ile rapor oluşturan kullanıcılar arasındaki ara katmandır. Biz de burada DWH\_OTO universe’ünü seçiyoruz.



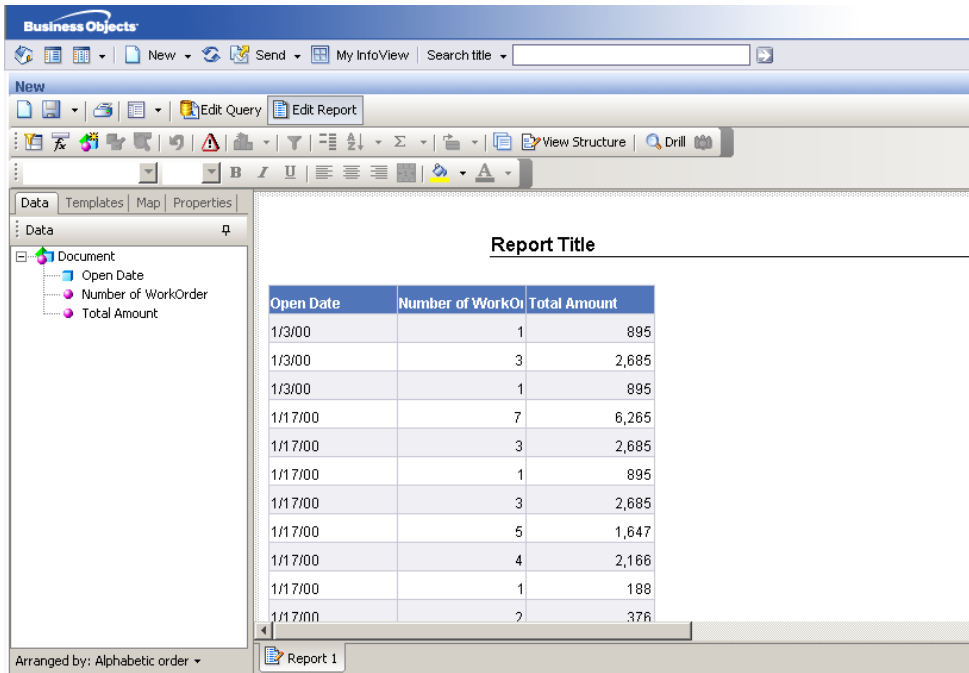
**Şekil 42: Business Objects XI Web Intelligence Universe Seçim Ekranı**

DWH\_OTO universe’ünü seçtikten sonra karşımıza gelen ekrandan raporda görmek istediğimiz nesnelere seçiyoruz. Burada görülen sol pencere universe’de oluşturulan objeleri içermektedir. Daha önce de belirttiğimiz gibi bu nesnelere aslında veri tabanındaki tabloların iş terimlerine çevrilmiş halidir. Biz burada “Open Date” yani açılış tarihi, “Number of Workorder” ve “Total Amount” nesnelere seçiyoruz. Filtre olarak da sadece 2000,2001 ve 2002 yıllarına ait bilgileri görmek istediğimizi belirtiyoruz. İstersek bu filtreyi dinamik olarak da oluşturabiliriz, prompt olarak adlandırılan bu dinamik filtrelemede raporun her yenilenmesinde kullanıcılara hangi yıllara ait bilgiyi görüntülemek istedikleri sorulacak ve sadece seçilen yıllar için rapor oluşturulacaktır.



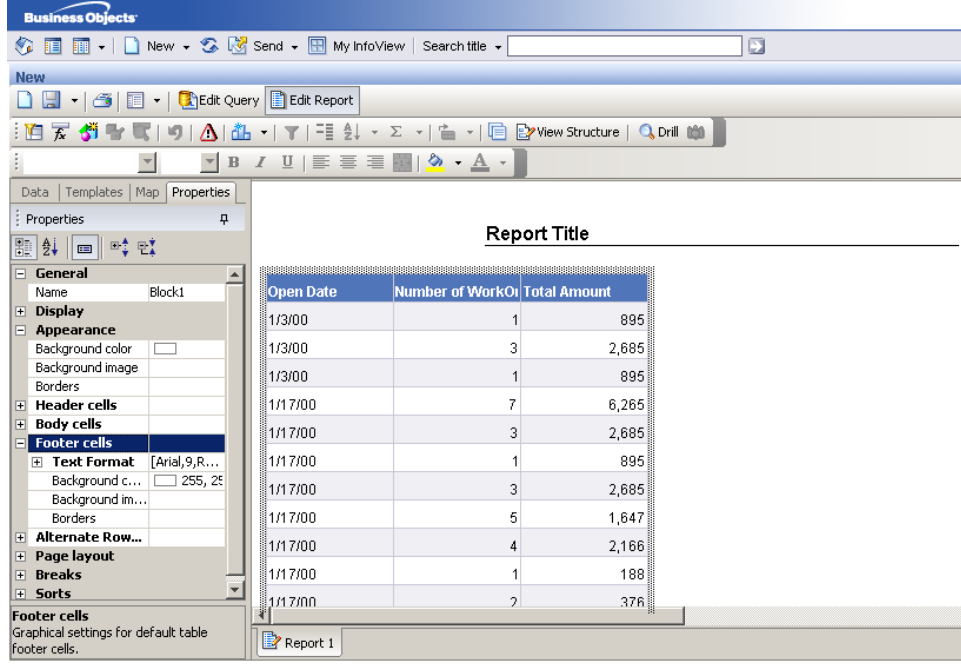
**Şekil 43 Business Objects XI Rapor Tasarlama Ekranı**

Seçilen nesnelere sonra oluşturulan sorgu çalıştırılır ve bizim belirlediğimiz nesnelere otomatik olarak bir tablo oluşturulur. Bu tabloya daha sonra yeni kolonlar eklenebilir veya çıkartılabilir, tekrar nesne seçim ara yüzüne dönülebilir ve yeni alanlar eklenebilir.



**Şekil 44: Business Objects XI Örnek Rapor Ekranı**

Tabloyu, kolonu veya satırı seçtikten sonra sol taraftaki pencereden Özellikler (Properties) bölümü seçilir ve burada seçilen alanın özellikleri görülebilir. Bu özellikleri dilediğiniz gibi değiştirebilir ve raporunuza istediğiniz tasarımı uygulayabilirsiniz.



Şekil 45: Business Objects XI Rapor Tasarlama Ekranı Özellikleri Araçları

### 3.3.2. Veri Ambarı Raporlar

Veri Ambarı oluşturma işlemi tamamlandıktan sonra sakladığımız verileri görsel olarak sunabilmek için artık Business Objects XI Web Intelligence modülü yardımıyla istediğimiz raporları Excel, Pdf ve Grafik gibi farklı formatlar da oluşturabilecek durumdayız. Şekil 46 da görüldüğü üzere Doğu Otomotiv Servis Ticaret A.Ş. Satış Sonrası Hizmetler bünyesinde Servis Hizmeti veren Doğu Oto Ankara, Doğu Oto Kartal ve Doğu Oto Maslak' a ait 2000 yılı İş Emri kapanış adetleri ve tutar bazında bir rapor hazırlamış bulunmaktayız. Raporun Şekil 46 da sade görünümü Şekil 47 ve Şekil 48 de ise Grafik sürümü rapor formatlarını hazırlamış bulunmaktayız.

Universe : Dogus D55 Firma İş Emri Analiz Report Filename : RPR003  
 Current User : Administrator Last Execution : 02.09.2007 10:49  
 Current Date : 02.09.2007 10:49 Last Modified : 02.09.2007 09:16

Seçilen Ay(lar) : Mayıs;Haziran;Temmuz;Ağustos  
 Seçilen Yıl(lar) : 2000  
 Seçilen Firmalar : Doğu Oto Ankara; Doğu Oto Kartal; Doğu Oto Maslak

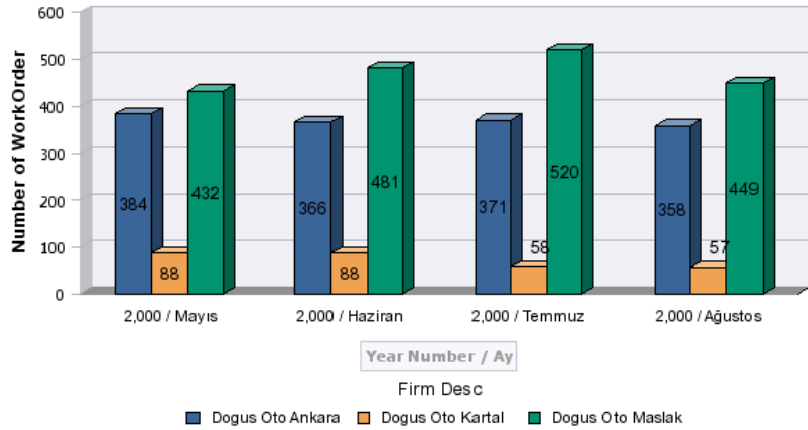
Firma İş Emri Analiz	2,000								2,000	
	Mayıs		Haziran		Temmuz		Ağustos		İş Emri Toplam	Tutar Toplam
	İş Emri	Tutar	İş Emri	Tutar	İş Emri	Tutar	İş Emri	Tutar		
Doğus Oto Ankara	384	79,680	366	160,122	371	277,137	358	169,416	1,479	686,355
Doğus Oto Kartal	88	64,650	88	42,240	58	27,840	57	27,360	291	162,090
Doğus Oto Maslak	432	270,939	481	343,434	520	366,300	449	241,005	1,882	1,221,678
<b>Toplam</b>	<b>904</b>	<b>415,269</b>	<b>935</b>	<b>545,796</b>	<b>949</b>	<b>671,277</b>	<b>864</b>	<b>437,781</b>	<b>3,652</b>	<b>2,070,123</b>

Şekil 46: Doğu Otomotiv Servis Ticaret A.Ş. 2000 Yılı SSH İş Emri Analiz Raporu

Şekil 47 ve Şekil 48 de ise Column Chart ve Pie Chart Grafik Raporlar daha çok üst düzey yönetim için hazırlanabilir. Kurumlarda yönetim kademesinde bulunan insanlar için görsel ağırlıklı grafik raporları yorumlamak daha kolaydır, renklendirilmiş barlar ve dilimler arasındaki farklılığı insan gözü ilk bakışta daha rahat algılar ve bu tarz raporların sunumu rakamsal ağırlıklı raporlara göre daha verimlidir.

Universe : Dogus D55 Firma İş Emri Analiz Report Filename : RPR003  
 Current User : Administrator Last Execution : 02.09.2007 10:49  
 Current Date : 02.09.2007 10:49 Last Modified : 02.09.2007 09:16

Seçilen Ay(lar) : Mayıs;Haziran;Temmuz;Ağustos  
 Seçilen Yıl(lar) : 2000  
 Seçilen Firmalar : Doğu Oto Ankara; Doğu Oto Kartal; Doğu Oto Maslak



Şekil 47: Doğu Otomotiv Servis Ticaret A.Ş. 2000 Yılı SSH İş Emri Analiz Raporu Column Chart Versiyonu



Universe : Dogus D55  
Current User : Administrator  
Current Date : 02.09.2007 10:49

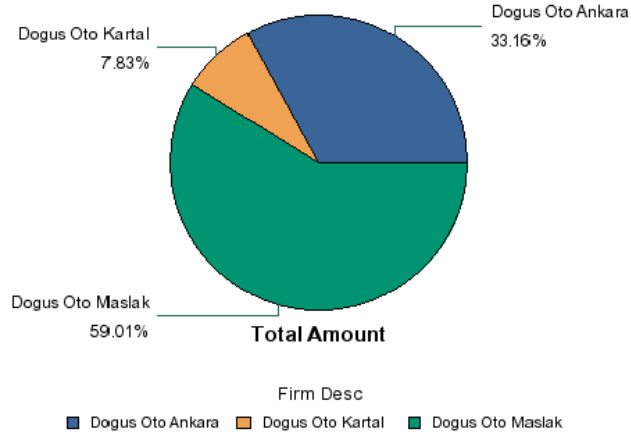
Firma İş Emri Analiz

Report Filename : RPR003  
Last Execution : 02.09.2007 10:49  
Last Modified : 02.09.2007 09:16

Seçilen Ay(lar) : Mayıs;Haziran;Temmuz;Ağustos

Seçilen Yıl(lar) : 2000

Seçilen Firmalar : Doğuş Oto Ankara; Doğuş Oto Kartal; Doğuş Oto Maslak



Şekil 48: Doğuş Otomotiv Servis Ticaret A.Ş. 2000 Yılı SSH İş Emri Analiz Raporu Pie Chart Versiyonu

## **Sonuç ve Genel Deęerlendirme**

Bu alıřmada Otomotiv pazarlama sektrnde faaliyet gsteren bir firma iin sektrn zellikleri ve firma st yneticilerinin bilgi gereksinimleri de dikkate alınarak bir veri ambarı modeli tasarlanmıř; bu tasarıma uygun biimde veri ambarı yaratılmıř; test edilmiř ve bazı ıktılar elde edilmiřtir.

Oluřturulan veri ambarı modeli mřterilerin ve araların bilgilerini iermekte, bu bilgilere dayalı olarak sorguların etkin biimde gerekleřtirilmesini saęlamaktadır. Veri ambarı “kar tanesi řeması” zerine dayandırılarak olgu tablosu ve boyutlar bu modele uygun biimde tanımlanmıřtır.

Veri ambarı oluřturulduktan sonra bu verilerle alıřmak zere bir Raporlama aracı kullanılmıř ve sorguların amacına ulařtıęı ve ok hızlı biimde alıřtıęı grlmřtir. Bylece bir veri ambarından beklenen, sorguların ok hızlı alıřması ve sorguların kolayca tanımlanması gibi zelliklere sahip olduęu anlařılmıřtır.

## **Kaynakça**

- Çarkıt, N., “Yönetim Süreçlerinde Bilgisayara Dayalı Karar Destek Sistemlerinin İrdelenmesi”, G.Ü. Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, Ankara,1997.
- Gray, Paul, Watson, J. Decision Support In The Data Warehouse, 1998.
- Harinarayan, V., Rajaraman, A., Ullman, J., D., Implementing data cubes efficiently. Int. Conf. Management of Data, Montreal, Kanada, Haziran 1996.
- Inmon, W., H., “What is a Data Warehouse?”, 1995.
- Inmon, W., H., Hackathorn, R., Using the Data Warehouse, 1994.
- Inmon, W., H., Building the Data Warehouse, 2005.
- Kimball, R., Ross, M., The Data Warehouse Toolkit, 2002.
- Kambayashi, Y., Mohania, M., Data Warehousing and Knowledge Discovery, İspanya, 2004.
- Microsoft Corporation, Microsoft Training And Certification Workbook
- Paul, L. George, L., “Oracle8,” Data Warehousing Guide, 1999.
- Robert, S., Joseph, C., Berkovitch, D. “Microsoft Data Warehousing,” Building Distributed Decision Support Systems, Mart, 1999.
- Srinivasan, A., Sundaram, D., Davis, J., Implementing Decision Support Systems: Methods, Techniques and Tools,Londra, 2000.
- Todman, C., Designing a Data Warehouse: Supporting Customer Relationship Management, 1997.
- Veri Madenciliği Eğitim Seminer Notu, Zeki Veri Madenciliği: Ham Veriden Altın Bilgiye Ulaşma Yöntemleri”,Bilişim, 2000.
- <http://www.bilisimsozlugu.com> 2007
- <http://www.dtm.gov.tr/dtadmin/upload/EAD/IstatistikDb/turkiyeoto.doc> 2006
- <http://www.dtm.gov.tr/dtadmin/upload/EAD/IstatistikDb/dun1.doc> 2006
- [http://www.dogusotomotiv.com.tr/icerik/108/364/dogus\\_otomotiv\\_hakkinda](http://www.dogusotomotiv.com.tr/icerik/108/364/dogus_otomotiv_hakkinda) 05.09.2007

## ÖZGEÇMİŞ

### 1. Bireysel Bilgiler

Adı : Erhan  
Soyadı : AKÖZ  
Doğum Yeri / Tarihi : İstanbul 1980  
Uyruğu : T.C.  
Adres / Telefon : Düğmeciler Caddesi. Damla Sitesi. C/4 Blok Daire:11 Eyüp  
İstanbul 0505 3579163

### 2. Eğitim

Beykent Üniversitesi  
Fen bilimleri Enstitüsü Bilgisayar Ağları ve İnternet Teknolojileri (2004–2007)

Beykent Üniversitesi  
İktisadi ve İdari Bilimler Fakültesi Yönetim Bilişim Sistemleri Bölümü (2001–2004)

Marmara Üniversitesi  
Teknik Bilimler Yüksek Okulu Bilgisayar Programcılığı (1998–2000)

Maçka Akif Tuncel Anadolu Teknik Lisesi  
Bilgisayar Bölümü (1994–1998)

### 3. Yabancı Dil

İngilizce; Çok iyi  
İtalyanca, Japonca; Başlangıç (Beykent Üniversitesi 2003–2004)

### 4. Ünvanlar

B.Sc, Beykent Üniversitesi Yönetim Bilişim Sistemleri, 2004

### 5. Mesleki Deneyim

Doğuş Otomotiv  
İnternet Yazılım Uzmanı (2000–2007)