

# VERİTABANI TASARIMI VE ÖRNEK GPS VERİTABANI

Özgür ÖZASLAN, Rahmi Nurhan ÇELİK\*

## ÖZET

*Verilerin organizasyonu, sunumu ve kullanılması bakımından önemli bir görev üstlenen Coğrafi Bilgi Sistemleri'nin temel bileşeni olan veri kadar, verilerin tutulma biçimleri de Coğrafi Bilgi Sistemlerinin sürdürülebilirliği ve etkin kullanımı açısından büyük önem taşımaktadır. Coğrafi Bilgi Sistemleri ile iç içe olan mesleğimizde veritabanlarının ve bu veritabanlarında verilerin tutulma biçimlerinin önemi her geçen gün daha da iyi anlaşılmaktadır. Bu çalışmada mesleğimizde günümüzün modern ölçme tekniklerinden birisi olan GPS ölçme tekniği kullanılarak elde edilmiş bilgi, ölçme ve değerlendirmelerin gerektiğinde tekrar kullanılabilmesi amacıyla ilişkisel bir veritabanı içerisinde nokta ve proje bazlı olarak nasıl tutulabileceği anlatılmıştır.*

## ABSTRACT

### DATABASE DESIGN AND SAMPLE GPS DATABASE

*Database design architecture, which is as important as data, is very important for effective usage and sustainability of Geographic Information Systems and it is also a fundamental component of Geographic Information Systems. Understanding of the importance of databases and database design architecture for GIS increases day by day in our profession. In this study managing GPS data based on control station and project information, that GPS is a modern surveying technique in our profession, for further use was explained.*

\*ozaslan@itu.edu.tr , celikn@itu.edu.tr

Jeodezi Anabilim Dalı, Jeodezi ve Fotogrametri Mühendisliği Bölümü  
İstanbul Teknik Üniversitesi, Ayazağa Yerleşkesi 80626 Maslak / İstanbul

## 1. GİRİŞ

Bilgi kadar bilginin saklanması fikri de oldukça eskidir. Çok eski tarihlerde bile insanların bilgileri, saklamak amacıyla kil tabletlere kazıdıkları bilinmektedir. Daha sonraları bilgiler kitaplarda tutulmaya başlanmış ve 20. yüzyılın ortalarında bilgisayarın kullanılmaya başlanmasıyla bilgisayar üzerinde tutulan veritabanlarının ve veritabanı yazılımlarının kullanılmasına geçilmiştir. Verilerin tutulma ve sunulma biçimleri zamanla değişmiş olsa da, veritabanlarının kullanılma nedeni dün de bugün de aynıdır. İnsanların sınırlı hatırlama ve akılda tutma kapasiteleri nedeniyle bilgileri bir veritabanında tutmak kaçınılmazdır. Ayrıca bilginin özel veya kamuya sunumu yani bilginin paylaşımı ve bu bilgilere hızlı, doğru,eksiksiz erişim de veritabanlarıyla mümkün olmaktadır. (Kaynarca, 1994)

Bu çalışmada bir projede GPS ölçme tekniği kullanılarak yapılmış ölçmelere ilişkin bilgilerin hangilerinin ve ne şekilde tutulabileceği ele alınmıştır. Projeye ve GPS ölçmelerine ilişkin tutulması yeterli olan bilgiler ve bu bilgilerin birbirleriyle nasıl bir ilişki içerisinde tutulabileceği gösterilmiştir. Bu çalışmada veritabanı herhangi bir veritabanı yazılımında kullanılacak nitelikte kavramsal ve fiziksel olarak tasarlanmıştır.

## 2. VERİTABANI

Veritabanı Türk Dil Kurumu sözlüğünde "Bilgisayar kullanımında çözüme erişmek için işlenebilir duruma getirilmiş bilgi ortamı" olarak tanımlanmaktadır. Bilişim dünyasında ise bu tanımlama, ufak değişiklikler göstermekle beraber, "Büyük boyuttaki verilerin gerektiğinde hızlı bir şekilde sorgulanabilmelerini ve kalıcılığını sağlamak amacıyla bilgisayar ortamında oluşturulmuş bir tür koleksiyondur." şeklindedir. (Gündüz, 2002)

Aşağıda veritabanı ile ilişkili temel bazı kavram ve tanımlara yer verilmiştir. Bu kavram ve tanımlardan sonra veritabanının ne olduğu daha net olarak anlaşılacaktır.

**Veritabanı Nelerden Oluşur :** Bir veritabanı ;

- Kolonlar ( Alanlar )
- Tablolar
- Birincil anahtarlar (Primary keys)
- Şemalar

gibi bölümlerden oluşur.

Kolonlar, veri tabanında bilgi içeren sütunlardır. Her kolon bir bilgi içerir ve içerdikleri bilgilere göre sınıflandırılır. Kolonlar bir araya gelerek tabloları oluştururlar. Tablodaki her bir satır ise ayrı bir veriyi gösterir . Birincil anahtar (primary key) tablodaki her bir satırın tekliğini tanımlar. Bilgilere hızlı ulaşmayı ve tablolar arasında bağlantı kurmayı sağlar. Birincil

anahtar olarak genellikle ID kolonları seçilir. Bir ya da daha çok kolon aynı anda birincil anahtar olarak seçilebilir, bu amaca göre değişir. Bir tablodaki birincil anahtarın diğer bir tabloda eşleştiği kolona yabancı anahtar (foreign key) denir. Ayrıca yabancı anahtar kayıtlı olmayan bilgiyi girmemizi engeller. Veri tabanı içindeki tabloların yerleştirme ve biçimlemesine şema denir. Şemalar aslında veritabanı yapısının görsel olarak tasarlanmış durumlarıdır.

**Veritabanı Yönetim Sistemi :** Bir veritabanı yönetim sistemi ( DBMS: DataBase Management System) kütüphane, uygulamalar ve yardımcı programların birleşmesinden oluşur ve verilerin saklanması, yönetilmesi ile ilgili konulardaki ayrıntıların daha kolay yapılabilmesini sağlarlar. VTYS ile kayıtların güncellenmesi ve kayıtlar üzerinde araştırma yapılması da mümkündür. Veritabanı Yönetim Sistemleri ( VTYS ) zamanla çok daha fazla özellikler kazanmaktadırlar. (Gündüz, 2002)

**Neden Veritabanı :** Veritabanları, üstte de belirtildiği gibi, verilerin saklanması ve yönetilmesi için kullanılmalıdır. Küçük bilgiler için kitaplar veya bilgisayarda kullanılan metin dosyaları yeterli olacaksa, bunun için veritabanı kullanılması gerekmez. Veri sayısı ve bu verilerin boyutu arttığında ve buna ek olarak hızlı ulaşım, aynı anda çoklu ulaşım, kolay güncellenebilirlik ve kişiye göre erişim hakkı gibi konular da söz konusu olduğunda veritabanlarını kullanmak kaçınılmazdır. Veritabanlarının başlıca kullanım nedenlerini maddeler halinde gösterecek olursak :

- Kayıt ekleme ve güncelleme kolaylığı,
- Sorgulama kolaylığı,
- İstenilen yapıda çıktı alabilme kolaylığı,
- Yedekleme kolaylığı,
- Verileri paylaşım kolaylığı,
- Verilere ulaşabilen kişi sayısının arttırılabilmesi,
- Verilere ulaşımın kişilere ve gruplara göre sınırlandırılabilmesi,
- İstatistiksel analiz yapma olanağı,
- VTYS ile birlikte verilerin saklanması ve yönetilmesi ile ilgili konulardaki ayrıntılardan veritabanı yöneticilerini kurtarması,
- Yukarıda belirtilen bütün işlemlerin diğer veri saklama yöntemleri ele alındığında oldukça hızlı yapılabilmesi,

şeklinde özetlenebilir. (Özaslan, 2002)

**İlişkisel Veritabanı :** Veritabanı yönetim sistemleri teorisi E. F. Codd'un 1970'de yazmış olduğu "A Relational Model of Data for Large Shared Data Banks" makalesi ile büyük bir ilerleme kaydetmiştir. Veritabanı tasarımı ve yönetimi alanında bir devrim niteliğindeki yazıda, Codd ilişki mantığını tanıtmış, tabloların gerçek dünyadaki nesnelere göstermekte ve bunların niteliklerini tutmakta nasıl kullanılabileceğini belirtmiştir. (Gündüz, 2002)

İlişkisel veritabanlarını, tablo yapısında herhangi bir düzenleme veya değişiklik yapmadan üzerinde her türlü sorgulama, güncelleme ve kayıt silme işlemini kolaylıkla ve etkin bir biçimde gerçekleştirebildiğimiz bir veritabanı yapısı olarak adlandırabiliriz.

İlişkisel veritabanlarında tablolardaki kayıtlar sıralı gruplar şeklindeki satırlar olarak tutulurlar ve bu satırlardaki her grup bileşeni aynı tipte veri içeren sütunları oluştururlar. Ayrıca bir tabloda bir veri iki defa tekrarlanmaz. Bir kaydın tekrarlanması gerekliliği ilişkisel yapı kullanılarak veritabanındaki diğer tablolarla, gerektiğinde de yeni ara tablolar kullanılarak çözümlür. İlişkisel veritabanlarının önemli bir diğer özelliği de işlem kolaylığı kadar veritabanına yeni veri gruplarının eklenebilmesi kolaylığıdır. Tüm veritabanını veya veritabanını yönetmek için yazılmış uygulamaların tamamını değiştirmeden yeni veri gruplarını veritabanına kolaylıkla eklenebilir.

İlişkisel veritabanında uygulama geliştiren programcılar, yazdıkları kodun veritabanının bütünlüğünü bozmamasına dikkat etmelidirler. Uygun bir örnek, bir müşteri kaydının silinmesi olabilir. Eğer müşteri tablosundan bir kayıt silinecekse, o kişinin siparis tablosundaki siparişlerinin de silinmesi gerekir. Aksi takdirde olmayan bir müşteriye ait siparişler veritabanında bulunacaktır. (Gündüz, 2002)

İlişkisel veritabanlarında kullanıcı ve program arayüzü olarak SQL (Structured Query Language) kullanılmaktadır.

**SQL :** SQL, veritabanı uygulamalarında, veri tanımlama, veritabanının bütünlüğünün kontrolü, veritabanlarına erişimin kontrolü veritabanlarının sorgulanması ve güncellenmesi amaçları için gerekli komutlara sahip olan bir alt dildir. Alt dil denilmesinin nedeni, bir bilgisayar dilinin sahip olması gereken tüm komutlara sahip olmayışından dolayıdır. SQL İngilizce’deki ‘Structured Query Language’ sözcüklerinin baş harflerinden oluşur. SQL’nin sahip olmadığı komutlar, çevrim (döngü) oluşturan komutlarla, if, then, else yada go to gibi kontrol ve dallanma komutlarıdır. Fakat SQL’nin diğer dillerle birlikte kullanılması mümkün olduğundan, gerekiyorsa SQL komutları diğer dillerin çevrim ya da kontrol komutları içerisinde kullanılabilir. (Özaslan, 2002)

### 3. VERİTABANI TASARIMI

Veritabanı tasarımı bir veritabanının kullanılmaya başlanması için atılmış ilk adımdır ve veritabanının sürdürülebilirliğinin sağlanmasında büyük önem taşır. İyi tasarlanmış bir veritabanında kayıt ekleme, silme, sorgulama, güncelleme gibi işlemler daha etkin olarak yapılabilir, veritabanına yeni veri gruplarının eklenmesi daha kolay olur. Veritabanı tasarımı genelde aşağıda sayılan yedi aşama ile gerçekleştirilir.

- Tutulması istenen verilerin belirlenmesi
- Verilerin niteliklerine göre gruplandırılması

- "Kavramsal Veritabanı Modeli"nin oluşturulması
- Veritabanındaki ilişkilerin yapısının belirlenmesi
- "Fiziksel Veritabanı Modeli"nin Oluşturulması
- "Fiziksel Veritabanı Modeli"nin gözden geçirilmesi
- Veritabanının oluşturulması

Tutulması İstenen Verilerin Belirlenmesi : Veritabanı tasarımına başlanmadan önce hangi bilgilerin veritabanında tutulacağı belirlenir. Bilgilerin ne amaçla tutulacağı da göz önüne alınarak bu belirleme yapılır. Belirlenen amacı karşılamaya yeterli olacak bilgiler bu belirlemede göz önüne alınmalı ve gereksiz bilgiler veritabanında tutulmamalıdır. Örnek vermek gerekirse veritabanında tuttuğumuz bilgilerden çıkarılabilecek başka bir bilgi ayrıca veritabanında tutulmamalıdır.

**Verilerin Niteliklerine Göre Gruplandırılması:** Tutulması istenen veriler belirlendikten sonra bu veriler niteliklerine göre gruplandırılırlar. Örnek vermek gerekirse bir kütüphane veritabanında kitaba ilişkin tutulması istenen tüm bilgiler bir grup, yazara ilişkin olarak tutulması istenen bilgiler ise başka bir grup oluşturacaktır. Bu aşama bazı kaynaklarda verilerin boyutlandırılması olarak da adlandırılmaktadır (kitap boyutu, yazar boyutu gibi). Burada dikkat edilmesi gereken nokta şudur: Bir gruba ilişkin tüm verilerin aynı tabloda tutulması gibi bir zorunluluk yoktur. Kavramsal veritabanını oluşturmaya ve tablo yapısını kurmaya atlık oluşturması ve tutulmak istenen bilgilerin atlanmaması için böyle bir boyutlandırma yapılır.

**Kavramsal Veritabanı Modelinin Oluşturulması :** Veriler belirlenip, niteliklerine göre gruplandırıldıktan sonra aralarında kurulacak olan ilişkiler de göz önüne alınarak "Kavramsal Veritabanı Modeli" oluşturulur. Kavramsal modelde tablo yapısı, tablolarda tutulacak verilerin neler olacağı ve bu tabloların hangilerinin birbirleriyle ilişkili olacağı belirlenir. Tablolardaki veri alanlarına girilecek veri tipleri ve uzunlukları, hangi alanlara anahtarlar konacağı da bu aşamada belirlenir. Bu model elle kağıt üzerinde oluşturulabileceği gibi, bu iş için hazırlanmış tasarım programlarını kullanmak da mümkündür.

**Veritabanındaki İlişkilerin Yapısının Belirlenmesi:** Kavramsal modelde hangi tabloların hangileri ile ilişkili olacağı belirlenir. Bundan sonra yapılacak olan kavramsal model üzerindeki tablolar arası ilişkilerin yapısının belirlenmesidir. Bu ilişkiyel yapı 4 şekilde olabilir.

Bire- bir : bir ana tablo bir alt tabloyu karşılar.

Bire- çok: bir tablo iki veya daha çok alt tabloyu karşılar.

Çoka-bir: iki ya da daha çok ana tablo bir alt tabloyu karşılar.

Çoka-çok: iki yada daha çok ana tablo iki yada daha çok ana tabloyu karşılar.

**Fiziksel Veritabanı Modelinin Oluşturulması:** Kavramsal model oluşturulduktan sonra fiziksel veritabanı modeli oluşturulur. Kavramsal veritabanı tasarımında bir bilgisayar programı kullanılmışsa, fiziksel yapı bu program tarafından otomatikman ve kavramsal yapı kontrol edilerek oluşturulabilecektir. Aksi takdirde kavramsal modeldeki tablolar ve bunlar ara-

sındaki ilişkisel yapı göz önüne alınarak tasarımcı tarafından fiziksel model kağıt üzerinde hazırlanacaktır. Fiziksel model, veritabanının veritabanı yazılımında tutulacak halidir. İlişkiler sonucunda fiziksel modelde kavramsal modelde oluşturulan tablolara ek tablolar oluşması mümkündür. Ayrıca yine ilişkisel yapı sonucunda bazı tablolardaki birincil anahtarlar, ilişkili oldukları tablolara yabancı anahtar olarak ekleneceklerdir ve tablo yapısı bu şekilde son halini alacaktır.

**Fiziksel Veritabanı Modelinin Düzeltilmesi:** Hazırlanan fiziksel model veritabanına aktarılmadan önce son bir kez gözden geçirilerek, gerek ilişkisel yapıda gerekse tablo içi veri gruplamasında veya veri türü, boyutu seçiminde bir yanlışlık olup olmadığı kontrol edilmelidir. Her ne kadar ilişkisel veritabanlarında veritabanına yeni veri grupları eklemek ve çıkarmak mümkün ve kolay olsa da, fiziksel modelde veya öncesindeki kavramsal modelde yapılmış büyük bir tasarım yanlışlığı veritabanının tamamen değişmesine neden olabilecektir.

**Veritabanının Oluşturulması:** Fiziksel veritabanı son halini aldıktan sonra yapılacak işi veritabanının ilgili yazılımda oluşturulması olacaktır. Veritabanı tasarım programlarında oluşturulan modelin SQL komutlarına dönüştürülmesi oldukça kolaydır ve oluşturulan bu SQL komut dosyası ile tek aşamada fiziksel model veritabanı yazılımında oluşturulabilir. Böyle bir program kullanılmamışsa, SQL komutları kullanılarak fiziksel model veritabanı yazılımında oluşturulur ve veritabanı kullanıma hazır hale getirilmiş olur.

#### 4. ÖRNEK GPS VERİTABANI TASARIMI

Tasarlanan GPS veritabanında bir projede gerçekleştirilen GPS ölçülerine ilişkin bilginin gerektiğinde daha sonra kullanılabilmesi için, nokta ve proje bazlı olarak tutulması öngörülmüştür. Tasarım için gerekli ön çalışma önce kağıt üzerinde yapıldıktan sonra, "Qdesigner" veritabanı tasarım programı kullanılarak kavramsal ve fiziksel model tasarlanmıştır.

**Tutulması İstenen Verilerin Belirlenmesi:** Bir projede yapılan GPS ölçmelerine ilişkin hangi bilgilerin ilerideki ölçme ve değerlendirmelerde kullanılmak üzere yeterli olacağı göz önüne alınarak, tutulacak veriler belirlenmiştir. Bu verilerin ne olduğu 4.2 aşamasında boyutlandırılmış (gruplandırılmış) olarak gösterilmektedir.

**Verilerin Niteliklerine Göre Gruplandırılması:** Gruplandırılmış verilerin ne olduğu Tablo 1 'de gösterilmiştir.

**Kavramsal Veritabanı Modelinin Oluşturulması :** Verilerin gruplandırılması yapıldıktan sonra, ilgili veri gruplarının tablolarda nasıl tutulacağı ve bu tablolar arası ilişkilerin nasıl olacağı belirlenmiştir. Kağıt üzerinde yapılan bir ön çalışmadan sonra "Qdesigner" yazılımında tasarıma devam edilmiştir. Kavramsal modelde tablo yapısı belirlenirken veritabanının en etkin nasıl kullanılabileceği ve veritabanında veri tekrarı yapılmadan bilgilerin nasıl tutulabileceği göz önüne alınmıştır. "Qdesigner" yazılımında oluşturulan kavramsal model, tablolar, veri tipleri ve "Qdesigner" yazılımını tasarım ekranı görüntüsü Şekil 1'de gösterilmiştir.

Proje	Kurum	Personel	İstasyon
İl İlçe Bölge Başlangıç Tarihi Bitiş Tarihi Notlar	Adı Adres Telefon Faks E-posta İlgili Kişi	Sicil no Kurum Kurumdaki Birim Adı Soyadı Adres E-posta Telefon Faks Notlar	No Adı X Y Z Gravite Topoğrafik Düzeltme Tesis Tarihi Tesis Türü Pafta Adı Ulaşım Bilgisi Kroki Röper Hesaplanma Yöntemi Diğer Projelerde Hesaplanmış Koordinatları ve Hesaplanma Yöntemi Notlar
Alet	GPS Oturum	GPS Değerlendirme	
Seri No Adı Türü	Alet Yüksekliği Kayıt Çizelgesi Veri aralığı Elevation Angle Observation Dosyası Navigation Dosyası Meteorology Dosyası	Dengeleme Sonuçları Baz Çözümleri	

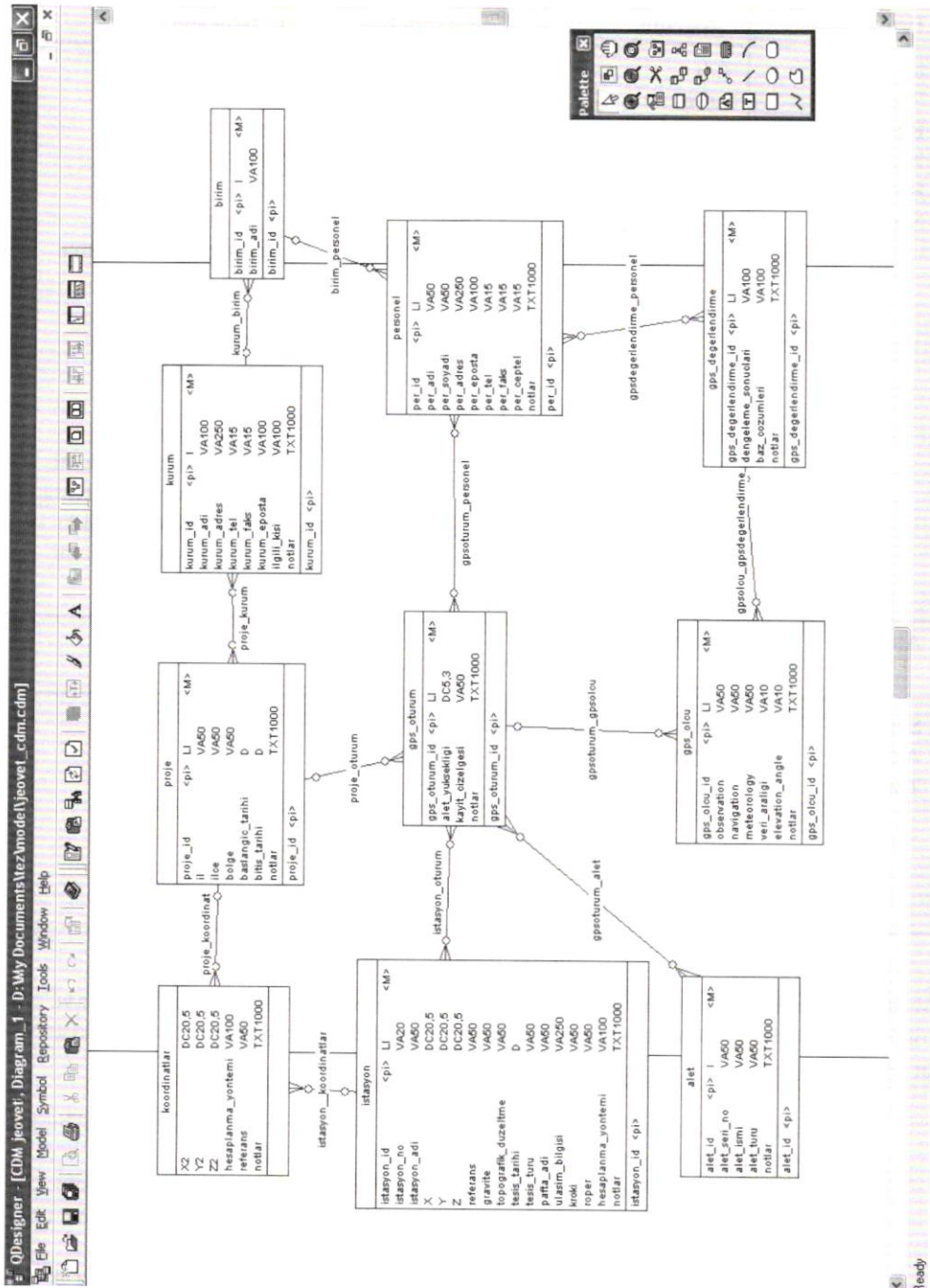
Tablo 1 : Verilerin Niteliklerine Göre Gruplandırılması

Veritabanındaki İlişkilerin Yapısının Belirlenmesi: 4.3 aşamasında belirlenen tablolar ve tablolar arası ilişkiler göz önüne alınarak ilişkilerin yapısı belirlenmiştir. Belirlenen ilişkiyel yapı aşağıdaki gibidir.

**Proje-Kurum:** Çoka-Çok (Bir projede birden fazla kurumun ortak çalışabileceği, bir kurumun da birden fazla proje yapabileceği göz önüne alınmıştır.)

**Kurum-Birim:** Bire-Çok (Kuruma yeni birimler eklenebileceği olasılığı göz önüne alınarak, kurum boyutunda yer alan birim ayrı bir tabloda gösterilmiştir. Bir kurumda birden fazla birim olacağı göz önüne alınarak ilişki bire-çok olarak kurulmuştur.)

**Birim-Personel:** Bire-Çok (Bir birimde birden fazla personel olabileceği ve her personelin sadece bir birime ait olabileceği göz önüne alınmıştır.)



Şekil 1 : Kavramsal Veritabanı Modeli



**İstasyon-GPS Oturum:** Çoka-Çok (Bir istasyonda birden fazla oturum yapılabileceği ve bir oturumda birden fazla istasyon kullanılacağı göz önüne alınarak ilişki çoka-çok olarak belirlenmiştir.)

**İstasyon-Koordinatlar:** Bire-Çok (Bir istasyona ait değişik projelerde hesaplanmış koordinat bilgilerini girebilmek amacıyla koordinatlar adı altında ayrı bir tablo oluşturulmuş ve ilişki bire-çok olarak alınmıştır.)

**GPS Oturum-Alet:** Çoka-Çok (Bir oturumda birden fazla alet kullanılacağı ve bir aletin de birden fazla oturumda kullanılacağı göz önüne alınarak ilişki çoka-çok olarak belirlenmiştir.)

**GPS Oturum- GPS Ölçü:** Bire-Çok (Bir oturumda birden fazla istasyona ait ölçme dosyaları bulunacağından ve bir ölçme dosyası sadece bir oturuma ait olabileceğinden ilişki bire-çok olarak alınmıştır.)

**GPS Oturum-Alet:** Çoka-Çok (Bir oturumda birden çok alet kullanılabilmesi ve bir aletin de birden fazla oturumda kullanılabilmesi göz önüne alınarak ilişki çoka-çok olarak belirlenmiştir.)

**GPS Oturum-Personel:** Çoka-Çok (Bir oturumda birden çok personelin görev alacağı ve bir personelin de birden fazla oturumda görev yapacağı göz önüne alınarak ilişki çoka-çok olarak alınmıştır.)

**GPS Değerlendirme-GPS Ölçme:** Bire-Çok (Bir değerlendirmede birden çok ölçme dosyası kullanılacağı ve her ölçmenin bir değerlendirmeye ait olacağı göz önüne alınarak ilişki bire-çok olarak belirlenmiştir.)

**GPS Değerlendirme-Personel:** Çoka-Çok (Bir değerlendirmede birden çok personelin görev alacağı ve bir personelin de birden fazla değerlendirmede yer alacağı göz önüne alınarak ilişki çoka-çok olarak alınmıştır.)

**Fiziksel Veritabanı Modelinin Oluşturulması:** Kavramsal model Qdesigner yazılımı ile oluşturulduğundan, fiziksel model otomatik olarak yazılım tarafından oluşturulmuştur. Fiziksel model ve Qdesigner yazılımındaki ekran görüntüsü Şekil 2'de gösterilmektedir. Çoka-çok ilişkilerin sağlanması için ara tablolar olduğu Şekil 2'de görülmektedir. Örneğin istasyon proje çoka-çok ilişkisini sağlamak için istasyon\_proje tablosu otomatik olarak oluşturulmuş ve istasyon id ile proje id hücreleri bu tabloya yabancı anahtar olarak eklenmiştir.

**Fiziksel Veritabanı Modelinin Düzeltilmesi :** Fiziksel model veritabanı yazılımına aktarılmadan önce gözden geçirilerek, düzeltmeler yapılmıştır. Alet tablosuna girilen bilgiler, istasyon ile ilişkilendirilmediğinden ve bu tablodaki bilgilerin istasyon ile olan ilgisinin gerekli olacağı düşüncesiyle fiziksel modelde oluşan gpsoturum\_alet tablosu istasyon tablosuyla

ilişkilendirilmiş; bu tabloya istasyon\_id yabancı anahtar olarak eklenmiştir. Yapılan GPS ölçmelerinin hangi istasyona ait olacağı bilgisinin girilebilmesi için gps\_olcu tablosu istasyon tablosu ile ilişkilendirilmiş ve gps\_olcu tablosuna istasyon\_id yabancı anahtar olarak eklenmiştir. Oturumlarda görev alan personelin hangi istasyonda görev aldığı bilgisini tutabilmek amacıyla da GPS oturum ve personel tablolarını ilişkilendiren gpsoturum\_personel tablosu, istasyon tablosu ile ilişkilendirilerek istasyon\_id bilgisinin bu ara tabloya yabancı anahtar olarak eklenmesi sağlanmıştır.

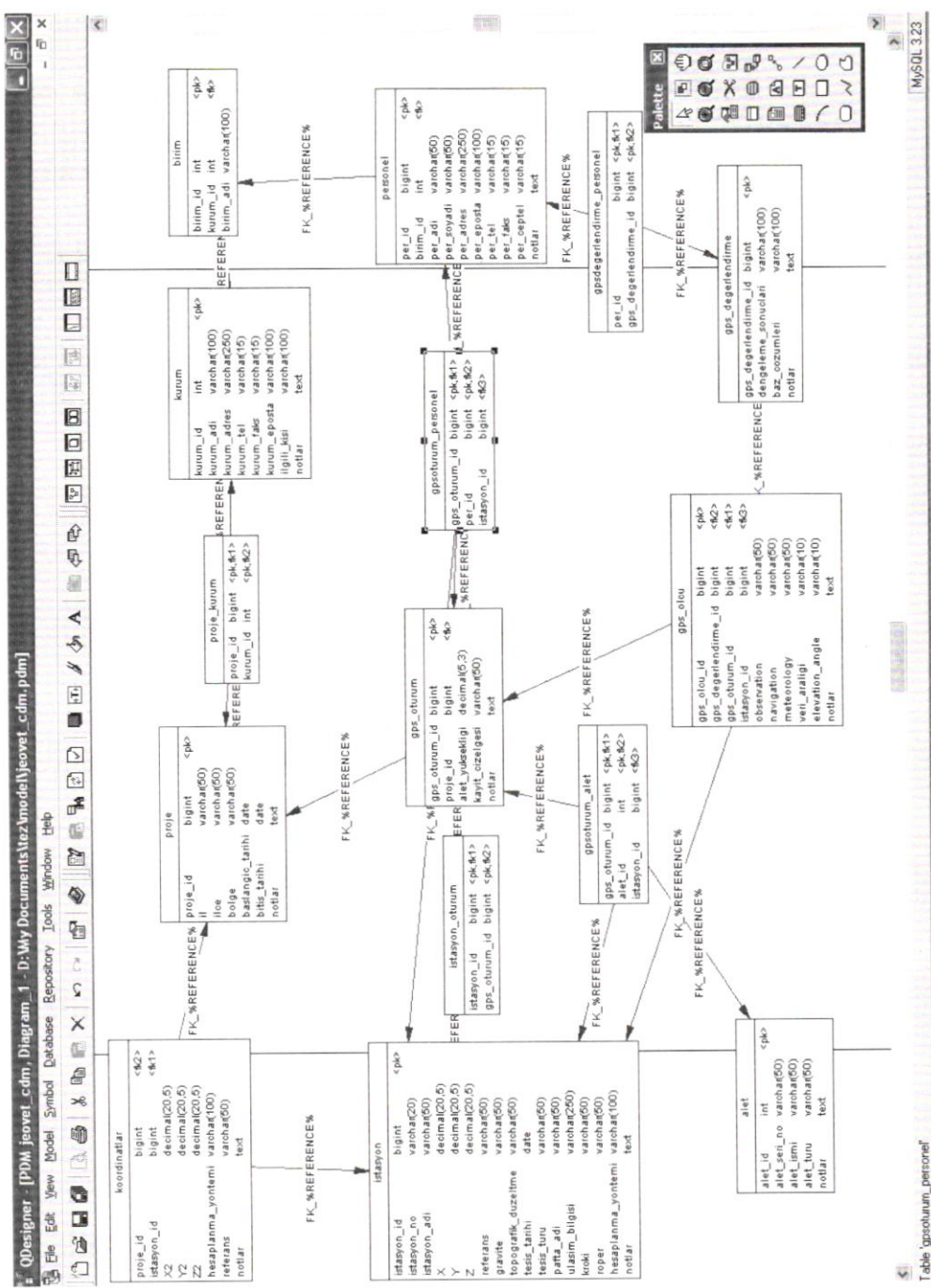
**Veritabanının Oluşturulması :** Son durumunu alan fiziksel veri tabanı Qdesigner yazılımı ile SQL komut dosyası haline dönüştürülmüştür. SQL komutlarını içeren bu dosya yardımıyla veritabanı, istenilen veritabanı yazılımında tek bir komut ile oluşturulabilir. Her veritabanı yazılımının SQL komut dosyalarını işleyebilme özelliği vardır. Aksi takdirde veritabanı tek tek SQL komutları ile, bir SQL komut arayüzü ile veya kullanacağınız veritabanına özel bir arayüz kullanılarak tasarımcı tarafından ayrıca oluşturulmak zorunda kalacaktır. (Bu çalışmada tasarlanan ilişkisel veritabanı MySQL veritabanı yazılımına aktarılmıştır.)

#### 4. SONUÇLAR ve ÖNERİLER

21. yüzyılda insanlığın elindeki en önemli silahlardan birisinin de bilgi olduğu unutulmamalıdır. Bilginin etkin bir silah olarak kullanılabilmesi ise ona doğru ve hızlı olarak ulaşılabilmesine bağlıdır. İşlenmeyen bir bilgiye ise hızlı olarak ulaşılması oldukça zordur. Bu bakımdan bilgi bir altın madenine benzetilebilir. Bu madende ne kadar altın olursa olsun, işlenmediği sürece gerçek değerine ulaşamayacaktır. Altınlar gerçek değerlerine; birileri onları bulup, çıkarıp, ayırıştırıp, insanların kullanımlarına sunduklarında ulaşmış olacaklardır. Günümüzde ise bilgiyi bu niteliklerde saklayabilmek veritabanlarıyla mümkün olabilmektedir.

Bu çalışma bilginin paylaşımı ve sürdürülebilirliğini sağlamakta önemli bir görev üstlenen veritabanlarının tasarımının bu sürdürülebilirlik ve etkin kullanımda ne kadar önemli olduğunu ortaya koymaktadır. Bu nedenle veritabanı tasarımı sağlıklı bir şekilde yapılmalı ve bu tasarım sırasında tasarım konusunda uzman kişiler ile ilgili veritabanını isteyen disiplinden ortak bir çalışma yürütülmelidir. İlişkisel yapı en başta sağlıklı kurulmalıdır. Aksi takdirde veritabanı tasarımında yapılacak hatalar ileride büyük sorunlara neden olabilir ve veritabanını etkin bir şekilde kullanımını olumsuz yönde etkileyebilir

Veritabanının tasarımı sırasında, en azından fiziksel veritabanı tasarımında, bir veritabanı tasarım programı kullanılması faydalıdır. Bu programlar; modeli kontrol etme, kaba hataları ortaya çıkarabilme ve sonrasında model SQL komut dosyalarına dönüştürerek tasarımcıyı fiziksel modelin oluşturulması sırasında gereksiz vakit kaybindan kurtarma özellikleri nedeniyle tercih edilmelidirler.



Şekil 2 : Fiziksel Veritabanı Modeli

## KAYNAKLAR

- ÇELİK, Rahmi N., ÖZLÜDEMİR M. Tevfik (1993), "Standart Veri Formatı Geliştirme ve Yararları", Prof. Dr. H. Wolf Jeodezi Sempozyumu, İstanbul, 3-5 Kasım 1993
- GÜNDÜZ, Devrim (2002), " Veritabanlarına Giriş Seminer Notları", Seminer, IV. Akademik Bilişim Konferansı, Konya
- KAYNARCA, Ayfer (1994), "A Geodetic Database Design Architecture", Yüksek Lisans Tezi, Jeodezi Anabilim Dalı, Kandilli Rasathanesi ve Deprem Araştırma Enstitüsü, Boğaziçi Üniversitesi, İstanbul
- ÖZASLAN, Özgür (2002), "Web Üzerinden Kütüphane Veritabanı Otomasyonu", Bitirme Ödevi, Mühendislik Bilimleri, Fen Edebiyat Fakültesi, İstanbul Teknik Üniversitesi, İstanbul
- Anonim, "MySQL Resmi Web Sitesi", <http://www.mysql.com>, 12.12.2002