

57762

KARADENİZ TEKNİK ÜNİVERSİTESİ FEN BİLİMLERİ ENSTİTÜSÜ

JEODEZİ VE FOTOGRAMETRİ MÜHENDİSLİĞİ ANABİLİM DALI

JEODEZİ VE FOTOGRAMETRİ MÜHENDİSLİĞİ PROGRAMI

ULUSAL KONUMSAL VERİ ALTYAPISI İÇİN

VERİ DEĞİŞİM STANDARDININ BELİRLENMESİ

Hrt. Yük. Müh. Çetin CÖMERT

Karadeniz Teknik Üniversitesi Fen Bilimleri Enstitüsünde

" Doktor "

Ünvanı Verilmesi için Kabul Edilen Tezdir

K.Ü. YÜKSEKÖĞRETİM ENSTİTÜSÜ
DOKÜMANLARI

Tezin Enstitüye Verildiği Tarih : 26.01.1996

Tezin Sözlü Savunma Tarihi : 24.05.1996

Tez Danışmanı : Prof.Dr. Gürol BANGER

Jüri Üyesi : Prof. Dr. Doğan UÇAR

Jüri Üyesi : Doç.Dr. Tahsin YOMRALIOĞLU

Enstitü Müdürü : Prof. Dr. Fazlı ARSLAN

Ocak 1996

57762

TRABZON

ÖNSÖZ

Bu tezde, "Ulusal Konumsal Veri Altyapısı" (UKVA) için bir veri değişim standardı önerilmiştir. UKVA ülke genelinde, konumsal veri erişimi ve kullanımını çok önemli ölçüde iyileştirerek, konumsal veriye ulaşmada yaşanan geleneksel kargaşa ve kaynak israfını önleyecek ve ülke ekonomisi için çok önemli bir katkı sağlayacaktır. Türkiye, "Bilgi Çağını" yakalamak ve bir "Bilgi Toplumu" olmak istiyorsa, UKVA'yı bir an önce hayata geçirmelidir. Çünkü UKVA, daha uzun süreçte bütün veri türlerine yönelik olarak oluşturulacak bir "Ulusal Veri Altyapısı" (UVA)'nın da çok önemli bir bileşeni olacaktır. Bu tez ile, UKVA için ilk adım atılmış ve çok önemli bir gereksinim karşılanmış olmaktadır. UKVA'nın kurulması ve yaşatılması sürecinde çok önemli bir katkı sağlayacağına inandığım bu tezin, bu süreci bir an önce başlatmasını diliyorum.

Çalışmalarım süresince her türlü desteğini esirgemeyen, önerileriyle katkıda bulunan Prof.Dr. Gürol Banger'e teşekkürü bir borç bilirim. Kanada'da birlikte çalıştığım Prof.Dr. Y.C. Lee'ye, yardımlarını esirgemeyen, Dale Lutz'a, SAIF projesi yöneticisi Dr. Mark Sondheim'a, Dr. Richard Pascoe'ya, ve Larry Moore'a teşekkürlerimi sunarım. Ayrıca, çalışmalarım süresince desteklerini esirgemeyen bölümdeki hocalarıma ve mesai arkadaşlarıma teşekkür ederim.

Trabzon, Ocak 1996

Çetin CÖMERT

İÇİNDEKİLER

	<u>Sayfa No</u>
TÜRKÇE ÖZET	VI
İNGİLİZCE ÖZET	VII
ŞEKİL LİSTESİ	VIII
TABLO LİSTESİ	IX
1. GENEL BİLGİLER	1
1.1 Giriş	1
1.1.1 Tanımlar	1
1.1.1.1 Coğrafi Bilgi Sistemleri	1
1.1.1.2 Konumsal Veri Değişimi	3
1.1.2 Problem	5
1.1.2.1 Ulusal Konumsal Veri Altyapısı	5
1.1.2.2 Konumsal Veri Değişim Standardı Problemi	7
1.1.3 Tezin Amacı ve Katkısı	9
1.1.4 Yaklaşım ve Yapılan Çalışma	10
1.1.5 Tez Konusu ile İlgili Diğer Çalışmalar	12
1.1.5.1 Ulusal Çalışmalar	12
1.1.5.2 Uluslararası Çalışmalar	13
1.1.5.3 Türkiye'deki Çalışmalar	16
1.1.6 Tezin Düzenlemesi	16
1.2 Konumsal Veri Modelleri ve Veri Yapıları	17
1.2.1 Veri Modeli Kavramı	18
1.2.2 Konumsal Veri Modelleri	21
1.2.2.1 Kayıt-Tabanlı Konumsal Veri Modelleri	24
1.2.2.1.1 Hiyerarşik Veri Modeli	24
1.2.2.1.2 Ağ Veri Modeli	25
1.2.2.1.3 İlişkisel Veri Modeli	26
1.2.2.2 Nesne-Tabanlı Konumsal Veri Modelleri	27
1.2.2.2.1 Varlık-İlişki Modeli	29
1.2.2.2.2 Nesneye-Dayalı Veri Modeli	30
1.2.3 Fiziksel Veri Modelleri	33

1.2.3.1 Dosya Yapıları ve Erişim Teknikleri	34
1.2.3.2 Konumsal Veri Yapısı ve Veri Formatı Arasındaki İlişki	37
1.3 Konumsal Veri Değişimi	39
1.3.1 Konumsal Veri Değişiminin İçeriği	40
1.3.1.1 Çeviriler	42
1.3.1.1.1 Kodlama Standartları	48
1.3.1.1.1.1 ISO 8211	49
1.3.1.1.1.2 ASN.1 Kodlama Kuralları	50
1.3.1.2 İletişim	51
1.3.1.2.1 Arabirimler	52
1.3.1.2.2 Veri Transferi	53
1.3.1.2.3 İletişim Servisleri	53
1.3.2 Konumsal Veri Değişim Yöntemleri	54
1.3.2.1 Dolaylı Yöntem	55
1.3.2.2 Doğrudan Yöntem	60
1.4 Ulusal Konumsal Veri Altyapısı	64
1.4.1 Veri Paylaşımı	65
1.4.2 Ulusal Konumsal Veri Altyapısı Gereksinimi	67
1.4.3 Ulusal Konumsal Veri Altyapısı	69
1.4.3.1 Kurumlar	70
1.4.3.2 Veri Kaynakları	71
1.4.3.3 Teknoloji	71
1.4.3.4 Konumsal Veri tabanları	72
1.4.3.5 Veri Ağları	74
1.4.3.6 Politikalar ve Standartlar	75
1.4.3.7 Kullanıcılar	76
1.4.4 Ulusal Konumsal Veri Altyapısının Oluşturulma	77
1.4.4.1 Politik Gereksinimler	77
1.4.4.2 Yönetimsel Gereksinimler	78
1.4.4.3 Teknik Gereksinimler	78
1.4.4.4 Stratejik Plan	79
2. YAPILAN ÇALIŞMALAR	81
2.1 Veri Modeli Ölçütleri	81
2.1.1 Veri Modeli Tipi	82
2.1.2 Raster Veri	83
2.1.3 Kartografik Gösterim	86
2.1.4 Koordinatlar	87
2.1.5 Topoloji-Geometri Ayırımı	88

2.1.6 Karmaşık Nesneler	89
2.1.7 Nesne Paylaşımı	90
2.1.8 Meta Veri	91
2.1.9 Nesne-Tabanlı Veri Değişimi	92
2.1.10 Zamansal Veri	93
2.1.11 Multimedya	95
2.1.12 Genişletilebilirlik	95
2.2 Gerçekleştirim Ölçütleri	96
2.2.1 Profiller	97
2.2.2 Detay Kodlama	98
2.2.3 Kodlama Yöntemi	101
2.3 Değişim Ortamı Ölçütleri	103
2.3.1 Dokümantasyon	103
2.3.2 Servisler	104
2.3.3 Değişim Ortamı Araçları	105
3. BULGULAR	108
3.1 SAIF	109
3.1.1 Veri Modeli	110
3.1.1.1 Veri Modeli Tipi	110
3.1.1.2 Raster Veri	111
3.1.1.3 Kartografik Gösterim	112
3.1.1.4 Koordinatlar	112
3.1.1.5 Topoloji-Geometri Ayırımı	113
3.1.1.6 Karmaşık Nesneler	113
3.1.1.7 Nesne Paylaşımı	113
3.1.1.8 Meta Veri	114
3.1.1.9 Nesne-Tabanlı veri Değişimi	114
3.1.1.10 Zamansal Veri	114
3.1.1.11 Multimedya	115
3.1.1.12 Genişletilebilirlik	115
3.1.2 Gerçekleştirim	116
3.1.2.1 Profiller	116
3.1.2.2 Detay Kodlama	117
3.1.2.3 Kodlama Yöntemi	118
3.1.3 Değişim Ortamı	119
3.1.3.1 Dokümantasyon	120
3.1.3.2 Servisler	120
3.1.3.3 Değişim Ortamı Araçları	121

3.2 SDTS	122
3.2.1 Veri Modeli	123
3.2.1.1 Veri Modeli Tipi	125
3.2.1.2 Raster Veri	125
3.2.1.3 Kartografik Gösterim	125
3.2.1.4 Koordinatlar	126
3.2.1.5 Topoloji-Geometri Ayırımı	126
3.2.1.6 Karmaşık Nesneler	127
3.2.1.7 Nesne Paylaşımı	127
3.2.1.8 Meta Veri	127
3.2.1.9 Nesne Bazında veri Değişimi	128
3.2.1.10 Zamansal Veri	128
3.2.1.11 Multimedya	128
3.2.1.12 Genişletilebilirlik	128
3.2.2 Gerçekleştirim	129
3.2.2.1 Profiller	129
3.2.2.2 Detay Kodlama	130
3.2.2.3 Kodlama Yöntemi	131
3.2.3 Değişim Ortamı	131
3.2.3.1 Dokümantasyon	132
3.2.3.2 Servisler	132
3.2.3.3 Değişim Ortamı Araçları	132
3.3 DIGEST	133
3.3.1 Veri Modeli	133
3.3.1.1 Veri Modeli Tipi	134
3.3.1.2 Raster Veri	135
3.3.1.3 Kartografik Gösterim	136
3.3.1.4 Koordinatlar	136
3.3.1.5 Topoloji-Geometri Ayırımı	136
3.3.1.6 Karmaşık Nesneler	137
3.3.1.7 Nesne Paylaşımı	137
3.3.1.8 Meta Veri	137
3.3.1.9 Nesne Bazında Veri Değişimi	138
3.3.1.10 Zamansal Veri	138
3.3.1.11 Multimedya	138
3.3.1.12 Genişletilebilirlik	138
3.3.2 Gerçekleştirim	139
3.3.2.1 Profiller	139

3.3.2.2 Detay Kodlama	140
3.3.2.3 Kodlama Yöntemi	141
3.3.3 Değişim Ortamı	142
3.3.3.1 Dokümantasyon	143
3.3.3.2 Servisler	143
3.3.3.3 Değişim Ortamı Araçları	143
4. İRDELEME VE DEĞERLENDİRME	145
4.1 İrdeleme ve Değerlendirme	145
4.1.1 Veri Modeli	145
4.1.2 Gerçekleştirim	147
4.1.3 Değişim Ortamı	148
4.2 İrdelemeler	150
4.3.SAIF'in Uyarlanması	152
5. SONUÇLAR	154
6. ÖNERİLER	158
7. KAYNAKLAR	160
8. EKLER	168
9. ÖZGEÇMİŞ	171

ÖZET

Ulusal Konumsal Veri Altyapısı için Veri Değişim Standardının Belirlenmesi

Günümüz beklenti ve sorunlarının birbiriyle ilişkili ve bütünleşik doğası karşısında hızlı, sağlıklı, ve ekonomik çözümler için, konumsal verinin etkin bir biçimde paylaşılması kaçınılmazdır. Ulusal Konumsal Veri Altyapısı (UKVA), ülke düzeyinde konumsal veri erişim ve kullanımını önemli ölçüde iyileştirerek, etkin konumsal veri paylaşımını olanaklı kılacaktır. UKVA ile, yerel ve Ulusal düzeylerde çeşitli kamu kurumlarının, yerel yönetimlerin, özel sektörün, Üniversitelerin ve çeşitli kullanıcı gruplarının bilgi sunacağı ve/veya bilgi alacağı bir "Bilgi Pazarı" oluşturulacaktır. UKVA ülke genelinde, konumsal veriye ulaşmada yaşanan geleneksel kargaşa ve kaynak israfını önleyerek, ülke ekonomisi için çok önemli bir katkı sağlayacaktır. UKVA'nın kurulması ve yaşatılması için bir dizi politik, yönetsel, ve teknik gereksinimin karşılanması gerekir. Türkiye Bilgi Çağını yakalamak ve bir Bilgi Toplumu olmak istiyorsa, UKVA'yı bir an önce hayata geçirmelidir.

Bu tezde, UKVA'da farklı kullanıcı gereksinimlerini karşılayacak ve taraflar arasında etkin veri paylaşımına olanak tanıyacak bir konumsal veri değişim standardı önerilmiştir. Bunun için, Kanada'nın SAIF, A.B.D.'nin SDTS ve NATO'nun DIGEST standartları, bu tezde belirlenen değerlendirme ölçütlerine göre değerlendirilmiştir. Değerlendirme ölçütleri bir konumsal veri değişim standardını genellik, genişletilebilirlik, esneklik, donanım, ve iletişim ortamı bağımsızlığı ile, veri değişim ortamı açısından değerlendirmektedir. Bu değerlendirmeye göre, UKVA için en uygun standart SAIF tir. SAIF, daha kapsamlı ve genişletilebilir bir veri modeline dayanması, esnek, iletişim ortamından bağımsız ve detay kodlamada yeni bir yaklaşım getirmesi ile, çevirici geliştiriciler için çok daha geniş olanaklar sunması bakımından SDTS ve DIGEST'den daha üstündür. UKVA'da SAIF'in kullanılması, SDTS ve DIGEST'i geçersiz formatlar yapmayacaktır. SAIF, DIGEST ve SDTS ile uyumludur. UKVA için SAIF'in önerilmesi ile UKVA'da konumsal veri değişimi problemi yalnızca kısmen çözümlenmiş olmaktadır. Problemin tamamıyla çözümü için, seçilen standarda Türkiye'ye özel bir takım eklemeler yapılması, standart detay/öznitelik katalogları ve sembol kitaplıklarının geliştirilmesi gibi, daha bir dizi gereksinim karşılanmalıdır. Ancak bu tez ile, UKVA için ilk adım atılmış ve çok önemli bir teknik gereksinim karşılanmış olmaktadır.

Anahtar Kelimeler : Ulusal Konumsal Veri Altyapısı, Konumsal Veri Değişim Standardı.

SUMMARY

Determining a Data Interchange Standart for the National Spatial Data Infrastructure

The integrated and interdependent nature of the new challenges and issues have rendered data sharing indispensable for providing timely and economical solutions. The National Spatial Data Infrastructure (NSDI) will greatly enhance nationwide access and usage of spatial data. NSDI will provide a "Information Marketplace" where public and private sector, local governments, universities, and various user groups exchange information. Enabling an effective sharing of spatial data, the NSDI will make a valuable contribution to the Nation's economy. Turkey has to immediately start building the NSDI if she wants to form an "Information Society". Building the NSDI involves a number of political, technical and management issues. One of the technical requirements of the NSDI is a spatial data interchange standard which would meet varying needs of so many divergent groups of the NSDI.

In this thesis, a spatial data interchange standard which will enable a effective sharing of spatial data within NSDI is proposed. For this purpose, SDTS of the US, Canada's SAIF, and the NATO's DIGEST standards have been evaluated according to the evaluation criteria determined in this thesis. The evaluation criteria examines a spatial data interchange standard in terms of its data model, implementation, and interchange environment and looks for generality, extendability, flexibility, hardware and communication media independence, and programmer/user support for the interchange. As the result of the evaluation, SAIF has been found to be the most appropriate for the NSDI of Turkey. SAIF has a more general and extendable data model than both SDTS and DIGEST and has a much stronger support for the interchange environment. Choosing SAIF will not make SDTS and DIGEST invalid since all are harmonized now. SAIF has to be extended for Turkey-specific needs. In addition, catalogs for features/attributes and symbols have to be developed for handling the data interchange in the NSDI properly. The thesis is a very first step towards the NSDI and will hopefully make a valuable contribution in its building.

Keywords : National Spatial Data Infrastructure, Spatial Data Interchange Standart.

ŞEKİL LİSTESİ

Sayfa No

Şekil 1.	Coğrafi Bilgi Sistemi	2
Şekil 2.	Veri tabanı tasarımı	19
Şekil 3.	Üç şema mimarisi	21
Şekil 4.	Konumsal veri modellerinin bir sınıflaması	23
Şekil 5.	Bir kadastro haritası ve değişik gerçekleştirmeleri	28
Şekil 6.	Genelleştirme ve Özelleştirme Hiyerarşisi	32
Şekil 7.	Birleştirme ve ayrıştırma kavramları	32
Şekil 8.	Bir parsele ait kayıt	35
Şekil 9.	Bir toprak haritasının farklı veri yapıları kullanılarak gösterimi.. . . .	38
Şekil 10.	Bir veri dizisinde sözdizim ve anlam	40
Şekil 11.	Konumsal veri değişiminin içeriği	41
Şekil 12.	Veri modeli, şema ve gerçekleştirim farklılıkları	43
Şekil 13.	Sunucu ve alıcı tarafında farklı topolojik veri yapıları	44
Şekil 14.	Türkçe, BNF ve ASN.1 dillerinde bir harita tanımı	48
Şekil 15.	ISO 8211 (VTD) dosya yapısı	49
Şekil 16.	ASN.1 de etiketler	50
Şekil 17.	MACDIF formatının ASN.1 deki coğrafi bölge ve projeksiyon tanımı . .	51
Şekil 18.	Konumsal veri değişim yöntemleri	54
Şekil 19.	Profil kavramı	57
Şekil 20.	Derleyici ve yorumlayıcı yapısı	61
Şekil 21.	ND veri değişim ortamı	64
Şekil 22.	UKVA elemanları	69
Şekil 23.	UKVA kavramı	70
Şekil 24.	Nesne-tabanlı ve Katman-tabanlı konumsal veri modelleri	83
Şekil 25.	Raster veri temsili	85
Şekil 26.	Nesne paylaşımı kavramı	91
Şekil 27.	Hiyerarşik detay kodlama	100
Şekil 28.	ISO OSI modeli	102
Şekil 29.	SAIF'te GeographicObject kavramı	111
Şekil 30.	SAIF'te SpatialObject sınıfı	112
Şekil 31.	SpatioTemporalObject sınıfı ve diğer ilgili sınıflar	115

Şekil 32.	SAIF veri grubunun yapısı	119
Şekil 33.	SDTS'nin organizasyonu	123
Şekil 34.	SDTS'nin genel yapısı	124
Şekil 35.	SDTS modülleri	130
Şekil 36.	DIGEST veri modeli	134
Şekil 37.	DIGEST raster veri modeli	135
Şekil 38.	DIGEST veri değişim yapısı	140



TABLO LİSTESİ

Sayfa No

Tablo 1.	Geniş Alan Ağları (WAN) için Seçenekler	74
Tablo 2.	DIGEST, SAIF, ve SDTS'nin Değerlendirme ölçütlerine göre incelenmesinden elde edilen sonuçlar	149



1. GENEL BİLGİLER

1.1 Giriş

Bilgi Teknolojisinin (BT) diğer alanlarında olduğu gibi, Konumsal Veri Yönetiminde de bilgisayar destekli sistemlerin "açık", "bütünleşik", "dağıtılmış", "federe" vs. olmalarının gerekliliği uzunca bir zamandır dile getirilmektedir. Söz konusu sistemlerin temellerini oluşturan ilkeler, "görevlerin dağıtılması", "kaynakların paylaşımı", ve fakat "ortaklaşa işlem" dir. Bütün bunların arkasındaki itici güç ise sürekli değişen gereksinimlerin hızlı, kolay, ve ekonomik bir şekilde karşılanabilmesidir. "Kaynakların paylaşımı" ile kastedilen donanım, yazılım, ve verinin paylaşımıdır. Bu tezin ilgi alanı veri paylaşımıdır.

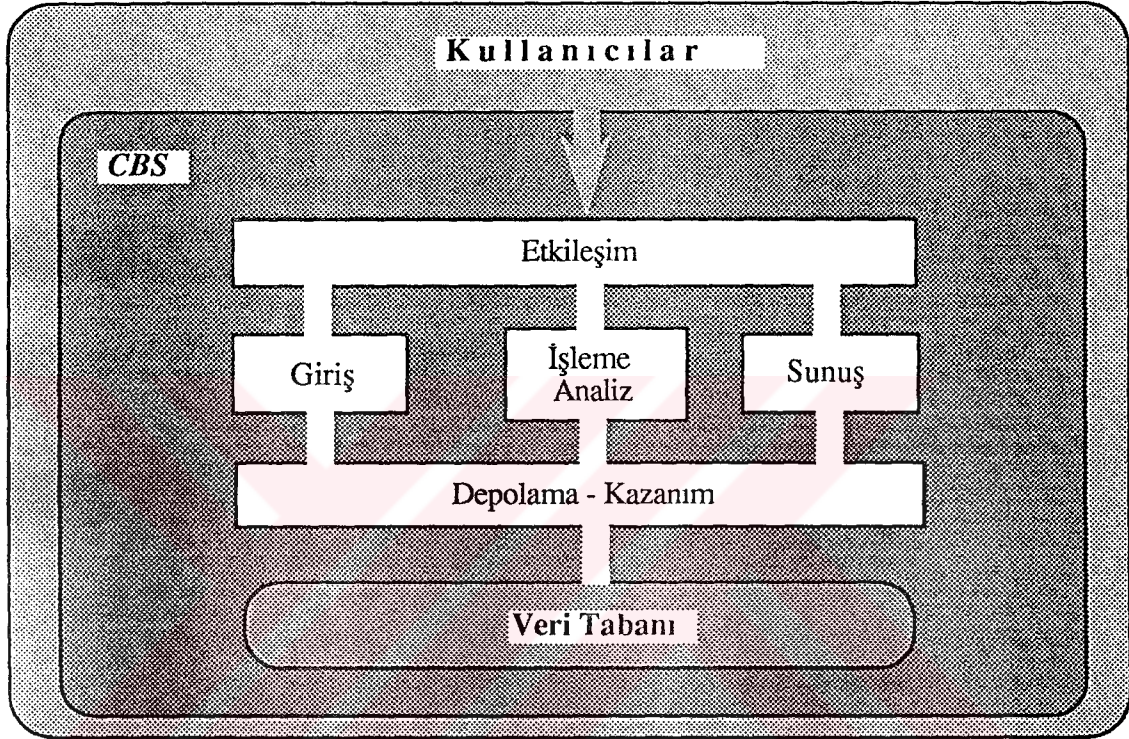
1.1.1 Tanımlar

1.1.1.1 Coğrafi Bilgi Sistemleri

Bilgi Teknolojisinin diğer alanlarında olduğu gibi, Coğrafi Bilgi Sistemleri alanında da yaşanan gelişmeler olağanüstüdür. Temelleri 1960'larda atılan Coğrafi Bilgi Sistemleri, bugün de çok yoğun ve giderek artan bir ilginin odağı durumundadır. Coğrafi Bilgi Sistemlerinin çıkış noktası, çok miktarda karmaşık verinin bilgisayar desteğinde, alışlagelmiş yöntemlerden çok daha etkin bir biçimde organizasyonu ve yönetimi ile, çok farklı gereksinimlerin hızlı, sağlıklı, ve ekonomik bir biçimde karşılanması olmuştur. İlk Coğrafi Bilgi Sistemi gerçekleştirimi olarak kabul edilen Kanada Coğrafi Bilgi Sistemi, 1960 larda çok büyük miktardaki Kanada arazi sayım verisini işlemek üzere geliştirilmişti [1], [2]. Coğrafi Bilgi Sistemleri bugün, mühendislikten pazarlamaya çok çeşitli alanlarda kullanılmakta ve giderek yaygınlaşmaktadır. Öyle ki, 1996 Dünya Coğrafi Bilgi Sistemi pazarının 10 milyar Dolar dolayında gerçekleşmesi beklenmektedir. Coğrafi Bilgi Sistemlerinin başlıca kullanım alanlarından bazıları yerel yönetim hizmetlerinin planlanması ve yönetimi, tesis planlama ve yönetimi, ulaşım planlama ve yönetimi, çevresel planlama ve değerlendirme, orman planlama ve yönetimidir [3].

Coğrafi Bilgi Sistemi (CBS) nin, standart olarak kabul edilmiş bir tanımı bulunmamaktadır [4], [5]. Genel olarak bir CBS, bir konumsal veri tabanı ve onun yönetimi için tasarlanmış yazılım ve donanım elemanlarının bir bütünüdür [6]. Bir CBS nin temel

fonksiyonları, konumsal verinin sisteme aktarılması, depolanması, işlenmesi ve analizi ile, bir formda kullanıcıya sunulmasıdır. [2], bir CBS de bu fonksiyonların yerine getirilmesi için, dört "alt sistem" tanımlamaktadır. Bunlar "giriş", "depolama ve kazanım", "işleme ve analiz", ve "sunuş" alt sistemleridir. Bunlara ayrıca, kullanıcının sistemle iletişimini sağlayan "etkileşim" alt sistemini de eklemek gerekir [7]. Şekil 1, bir CBS nin yapısını göstermektedir.



Şekil 1. Coğrafi Bilgi Sistemi.

Bu tanımlamaya göre, CBS işlevlerinden yalnızca bir veya bir kaçını yerine getirebilen Bilgisayar destekli tasarım ve harita üretim sistemleri, uzaktan algılama ve görüntü işleme sistemleri, sayısallaştırma sistemleri gibi sistemler, CBS olarak kabul edilemezler. Örneğin bir sayısallaştırma sistemi yalnızca, mevcut haritaların sayısallaştırılmasına yönelik, sınırlı veri depolama ve kazanım olanakları sunar. Ancak CBS dahil bütün bu sistemlerin ortak yanı, verinin bilgisayar desteğinde, sayısal olarak, alışlagelmiş yöntemlerden çok daha etkin bir biçimde çekip çevirilmesi ya da yönetimidir. Bu sistemlerin ilgi alanını oluşturan veri ise, "karmaşık" bir veri türü olarak nitelenen "konuma dayalı" veridir.

Konuma dayalı veri ile kastedilen, konumu bir şekilde tanımlanabilen veridir. Konum, belirli bir koordinat sisteminde koordinatlarla mutlak ya da bağıl olarak tanımlanabilir. Örneğin, bir parselin konumu iki boyutlu bir koordinat sistemindeki koordinatlarıyla ya da diğer bir parsel ile olan komşuluğu ile belirtilebilir. CBS ler alanında

standart bir terminolojinin eksikliği nedeniyle, bu tür veri için yabancı literatürde çok çeşitli isimler kullanılmaktadır. "Coğrafi veri", "kartoğrafik veri", "sayısal harita verisi", "coğrafi referanslı veri", "konum referanslı veri", ve "konumsal veri" bunlardan yalnızca bir kaçıdır. Bu tezde, bu tür veri "konumsal veri", bu verinin sayısal olarak yönetimini bir şekilde üstelenen sistemler de, Konumsal Veri Yönetim Sistemleri (KVYS) olarak anılacaktır.

1.1.1.2 Konumsal Veri Değişimi

CBS ler gerek ilk başta işlevsel hale geçişlerinde ve gerekse yaşamları boyunca sürekli olarak konumsal veriye gereksinim duyar. Gereksinim duyulan veri iki şekilde sağlanabilir. Bunlardan biri verinin ilk elden toplanması, diğeri ise sayısal formda başka bir kaynaktan sağlanmasıdır. Konumsal verinin ilk elden toplanması, kullanılan yöntem ne olursa olsun, oldukça pahalı ve zaman alıcı bir işlemdir. Öyle ki, bir CBS kurmada veri toplamanın toplam maliyete oranı % 60-80 olarak kabul edilmektedir [8], [9]. [10], CBS'nin yaşam sürecini de dikkate alarak donanım, yazılım, ve veri maliyetleri arasında 1:10:100 oranını vermektedir. Yüksek maliyet bir yana, verinin ilk elden toplanması, eldeki proje süresi açısından da kabul edilemez bir çözüm yolu olabilir. İlk elden veri toplamanın en uygun yanı, tamamıyla istenen özelliklerde veri elde edilmesidir. Başka bir kaynaktan sağlanacak olan veri, örneğin "doğruluk" açısından yetersiz olabilir.

Zaman ve maliyet açısından daha uygun olan seçenek, verinin hazır sayısal formda başka bir kaynaktan sağlanmasıdır. Konumsal veriye Mühendislikten Pazarlamaya, Çevre Korumadan Arkeolojiye çok geniş bir yelpazede gereksinim duyulması nedeniyle, aynı veri farklı sistemler/uygulamalar tarafından kullanılabilir, yani "paylaşılabilir". Örneğin hem tesis hatları yönetimi ve hem de ulaşım ağı planlamasında konumu tanımlayan veriye gereksinim vardır. Konumsal veri yönetiminde hiç te yeni bir yaklaşım olmayan "veri paylaşımı", CBS lerin ortaya çıkmasıyla önemini daha da artırmış, BT'de özellikle son on yıldaki gelişmelerle kaçınılmaz olmuştur; Farklı özelliklerde sayısal konumsal verinin artık çok çeşitli kaynaklardan sağlanabilmesi, CBS lerin mesafe tanımaksızın birbirleriyle çevrim-içi iletişim kurabilmeleri, konumsal veri yönetiminde yeni ufuklar açmış ve bir sistemin tek başına veri toplama kapasitesini oldukça aşan uygulama ve projeleri gündeme getirmiştir.

İki CBS arasında veri paylaşımının alışlagelmiş yolu, "konumsal veri değişimi"dir. Bir konumsal veri değişiminin tarafları bu tezde, "sunucu" ve "alıcı" olarak anılacaktır. Sunucu veriyi sağlayan, alıcı ise veriyi transfer eden taraftır. Konumsal veri değişimi yalnızca CBS lerle sınırlı değildir. Sunucu ve alıcı iki KVYS olabileceği gibi, alıcı bir KVYS, sunucu ise veri sağlayıcı bir kuruluş olabilir. Konumsal veri değişiminin iki boyutu

vardır. Bunlar "Çeviri" ve "iletişim" boyutlarıdır. İki farklı sayısal veri gösterimi arasındaki dönüşüm çeviri, verinin fiziksel olarak alıcıya iletilmesi de iletişim boyutu kapsamındadır.

Konumsal veri değişiminde çeviri, aynı bir gerçeğin, alıcı ve sunucu tarafında farklı şekillerde tanımlanmış olması nedeniyledir. "Gerçek" sözcüğü ile, bir KVYS'nin konusunu oluşturan gerçek dünya kastedilmektedir. Aynı anlamı karşılamak üzere tez içerisinde, "veri" sözcüğü de kullanılmaktadır. Farklı kullanıcılar aynı bir gerçeği ilgi alanlarına "en uygun" bir biçimde tanımlarlar. Örneğin, bir harita mühendisi ile bir ziraat mühendisi, aynı bir arazi parçasını gösterim eden bir harita üzerinde farklı bilgiler görmek isterler. Harita mühendisinin parsel tanımlamasında yer almayan "toprak cinsi", ziraat mühendisi için "olmazsa olmaz" bir özellik olabilir. Buna karşılık, arazinin topoğrafik yapısı her ikisi için de çok önemli olabilir.

Gerçeğin bir tanımlaması onun, tanımlamanın bakış açısından bir modelidir. Bir KVYS'de bu modelin kullanıcının ve bilgisayarın anlayış düzeylerinde tanımlanması gerekir. Bunun için "veri modelleri" kullanılır. Bir veri modelinin anlatım gücü içerdiği kavramlarla sınırlıdır. Örneğin "komşuluk" kavramını içermeyen bir veri modeli ile iki parselin birbirine komşu olması belirtilemeyecektir. Dolayısıyla kullanılan veri modeli kavramsal olarak ne kadar zengin ise, gerçek algılandığına o kadar yakın bir biçimde tanımlanabilecektir. Konumsal veri için çok farklı veri modellerinin kullanılması, konumsal veri değişimini oldukça karmaşık bir işlem yapmaktadır [11], [12].

Gerçeğin kullanıcı düzeyindeki tanımı "soyut" bir tanımdır. Gerçekleştirmede bu tanımlama bilgisayarın anlayacağı düzeye indirgenir ve "somut" bir biçimde ifade edilir. Konumsal veri değişimi somut tanım üzerinden gerçekleşir. Somut tanım da, soyut tanım gibi belirli bir söz dizimine uyar ve bu genellikle verinin "format"ı olarak anılır. Dolayısıyla, çeviri işleminin gerçekleştirilmesi için verinin her iki taraftaki sayısal gösterimlerinin soyut ve somut sözdizimlerinin bilinmesi gerekir. Soyut ve somut sözdizim kavramları 3. bölümde ayrıntılı olarak açıklanmaktadır.

Konumsal veri değişimi, bir formattan diğerine "doğrudan" ya da bir ara format üzerinden "dolaylı" olarak gerçekleştirilebilir. Genel amaçlı olarak geliştirilmiş ara formatlar, "değişim formatı" olarak anılır. Geleneksel olarak, bir değişim formatının bir soyut, bir de somut tanımı vardır. Soyut tanım formatın veri modeline işaret eder. Somut tanım ise, formata göre verinin fiziksel kodlama kurallarını belirler. Formatın donanım bağımsızlığının sağlanması için, soyut ve somut tanımlar birbirinden ayrı olarak düşünülmelidir. Dolaylı yöntemde iki sayısal veri gösterimi arasında "eksiksiz"—bilgi kaybı olmayan—bir çeviri için, değişim formatının her iki gösterimin veri modellerini tamamen kapsaması gerekir.

Örneğin zaman boyutunu içermeyen bir değişim formatı ile, zamansal veri transfer edilemeyecektir. Pratikte daha yaygın kullanım alanı bulmuş olan, dolaylı yöntemdir. Bununla birlikte, gerek teknik ve gerekse teknik olmayan nedenlerle, veri değişimi için herkesin kendi standardını tercih etmesi sonucu, konumsal veri değişiminde bugün çok çeşitli "standartlar" kullanılmaktadır. Bunun yarattığı en önemli sorun, bir sistemin farklı kaynaklardan veri transfer edebilmesi için çok sayıda formatı tanımasıdır.

Konumsal veri değişimi, bugüne kadar özel amaçlı çözümlerle ele alınmış olmasından ötürü, bugün arzu edilen bir düzeyde gerçekleştirilememektedir [11], [13], [14], [15], [16]. Bu da aslında, değişim formatı tasarımının hep ön planda tutulmuş olması, gerçekleştirim boyutunun ise adeta gözardı edilmiş olması nedeniyledir. Gerçi, değişim formatı tasarımı alanında önemli gelişmeler sağlanmıştır. Ancak, konumsal veri değişiminin pratikte uygulanışı ele alındığında, veri gruplarını görme, tarama, sorgulama ve ardından da yalnızca istenen alt grupların ya da detayların değişimine olanak tanıyacak kullanıcı arabirimlerinin eksikliği göze çarpmaktadır [15].

1.1.2 Problem

1.1.2.1 Ulusal Konumsal Veri Altyapısı (UKVA)

Günümüz beklenti ve sorunlarının birbiriyle ilişkili ve bütünselik doğası karşısında hızlı, sağlıklı, ve ekonomik çözümlere ancak, konumsal verinin "yatay" ve "düşey" doğrultularda etkin bir biçimde paylaşılması ile ulaşılabilir. Yatay doğrultuda veri paylaşımı ile kastedilen, farklı amaçlara hizmet eden ve farklı tipte veri ile çalışan kurumlar arasındaki veri paylaşımıdır. Harita Genel Komutanlığı (HGK)'nın bir tarımsal ürün rekolte tahmini çalışması için Devlet İstatistik Enstitüsü (DİE)'ne, DİE'nin de Tarım Bakanlığına veri sağlaması gibi. Düşey doğrultuda veri paylaşımı ise, verinin yerel, bölgesel, ulusal, ve hatta global düzeylerde paylaşımıdır. Bugün pek çok sorunun çözümü kurumların işlevsel ve yetkisel sınırlarını aşmaktadır. Örneğin yerel yönetimler son yıllarda en önemli sorunlarından biri haline gelen katı atık yönetimine yönelik uygun yer seçimi için, merkezi yönetimden veri istemek durumunda kalabilir. Ters yönde merkezi yönetim, zehirli artıkların yönetimi ile ilgili ulusal politikalar üretmek amacıyla gerekli verileri yerel yönetimlerden sağlamak durumundadır.

Konumsal veri ile çalışan kamu ve özel sektör kurumları arasında, gerek yatay ve gerekse düşey doğrultularda etkin veri paylaşımının sağlanabilmesi için, çeşitli kurum ya da kullanıcı gruplarının gereksinim duydukları veriye kolayca ulaşabilecekleri bir "Bilgi Otobanı"nın oluşturulması gerekmektedir. Bilgi otobanının hizmet sunduğu alan ne kadar

geniş olursa, sağlayacağı yarar da o derece büyük olacaktır. Buna karşılık bilgi otobanının hizmet alanı genişledikçe, kurulması ve yaşatılması ile ilgili teknik ve kurumsal sorunlar daha karmaşık bir hale gelebilir. Ülke genelinde hizmet sunacak bir bilgi otobanı, konumsal veriye ulaşmada yaşanan kargaşa ve kaynak israfını gidererek, ülke ekonomisi için çok önemli bir katkı sağlayacaktır. Son yıllarda kavramsallaştırılan Ulusal Konumsal Veri Altyapısı (UKVA) ile, ülke düzeyinde etkin konumsal veri paylaşımını olanaklı kılacak bilgi otobanlarının oluşturulması hedeflenmektedir. UKVA ile konumsal veriye erişim ve kullanım önemli ölçüde iyileşecek ve ülke genelinde yatay ve düşey veri paylaşımı sağlanabilecektir. UKVA ile, Ulusal düzeyde bir veri paylaşım programının gerek hayata geçirilmesi ve gerekse sürdürülmesinde karşılaşılabilecek teknik ve kurumsal problemler daha kapsamlı ve tutarlı bir çerçevede ele alınarak, çözüm yöntem ve kuralları belirlenecektir [17], [18], [19]. Son bir kaç yıldır, başta Amerika olmak üzere pek çok ülke, bir UKVA kurma çabası içerisinde [11], [19]. Öyle ki [19], UKVA'yı Amerika'nın Dünya pazarında ekonomik rekabet gücünü sürdürebilmesinde en önemli etken olarak görmektedir.

Türkiye'de bugün, konumsal veri yönetiminde CBS'lerin alışlagelmiş yöntemlerin yerini alması süreci yaşanmaktadır. CBS'ler Üniversiteler, kamu kurumları, yerel yönetimler, ve özel sektör için giderek daha da yoğunlaşan bir ilgi odağı durumundadır. Bir takım kamu kurumu ve belediyeler şimdiden CBS için gerekli yazılım ve donanım desteğine sahiptir. Diğer bir kısım, CBS'lere olan gereksinimin bilincinde somut adımlar atmaya planlamaktadır. Nihayet, özel sektör bu işin içindedir. Bununla birlikte Türkiye, CBS'lere geçiş sürecinin henüz çok başlarında ve bu alandaki öncü ülkelere göre oldukça geri kalmış durumdadır. Örneğin Kanada ve A.B.D., CBS çalışmalarına 1960'lı yıllarda başlamıştır [12], [13]. Bu ülkeler, konumsal veri yönetiminde bütün sorunlarını çözmüş olmamakla birlikte, veri paylaşımı alanında önemli bir noktaya gelmişlerdir. CBS ve veri paylaşımının birbirinden ayrılmaz kavramlar olmaları nedeniyle, CBS'lerin kullanıma girmeleri ile birlikte veri paylaşımı gündeme gelmiştir. Dolayısıyla, CBS öncü ülkeleri veri paylaşımı alanında çok geniş bir deneyime sahiptir. Bu ülkelerde çeşitli veri paylaşım programları 1970 ve 1980 li yıllarda uygulamaya konmuş ve bu programlarda kısmen başarı sağlanmıştır [12], [19]. Özetle Türkiye'nin yapması gereken, CBS'ler alanındaki geri kalmışlığını, CBS teknolojisinde öncü, bu teknolojinin uyarlanması sürecinde pek çok sorununu çözmüş ülkelerin deneyimlerinden yararlanarak bir an önce gidermektir.

Konumsal veri ile çalışan kurumların aynı veriyi paylaşmak yerine ilk elden toplama yoluna gitmeleri, Türkiye'de de geleneksel bir rahatsızlıktır. Benzer şekilde, konumsal veri ile iş gören pek çok kamu kurumunun gerek kendi içlerinde ve gerekse diğer kurumlarla ilişkilerindeki "hantallık", öteden beri şikayet konusudur. Türkiye, veri paylaşımı alanında CBS teknolojisinde öncü ülkelerin bugün ulaştığı noktaya, onların geçtiği engebeli yolları

aynen izleyerek ulaşmak zorunda değildir. Esasen Türkiye'nin buna tahammülü yoktur. O nedenle, Türkiye kendi UKVA'sını bir an önce kurmalıdır. UKVA ile aynı verinin farklı kurumlar tarafından ilk elden toplanması önlenecektir. UKVA çerçevesinde ayrıca, konumsal veri ile çalışan bir takım kamu kurumlarının "Bilgi Çağı"nın gerektirdiği dinamizmi yakalamaları yolunda çoktandır gereksinim duyulan "yeniden yapılanma"sı gerçekleştirilebilecektir. Çünkü UKVA, söz konusu kurumların işleyişi ve yapısı ile ilgili bir takım yasal düzenlemeleri gerektirecektir.

Türkiye Bilgi Çağını yakalamak ve bir Bilgi Toplumu olmak istiyorsa, UKVA'yı bir an önce hayata geçirmelidir. Çünkü UKVA, bütün veri türlerine yönelik olarak olarak oluşturulacak bir Ulusal Veri Altyapısı (UVA)'nın da çok önemli bir bileşeni olacaktır. "UKVA" Türkiye için oldukça yeni bir kavramdır. O nedenle UKVA için ilk adım, UKVA kavramının siyasi otoriteye, UKVA taraflarına, ve kamuoyuna anlatılarak, ulusal bir politika olarak benimsenmesini sağlamaktır. Kuşkusuz, UKVA öncelikle siyasi otorite tarafından desteklenmelidir. Ardından, UKVA taraflarının politik ve finansal desteklerinin kazanılması gerekir. Bunun için, UKVA'yı yürütecek yeni bir kurumsal yapılanma oluşturulmalıdır.

UKVA, yerel ve Ulusal düzeylerde çeşitli kamu kurumlarına, yerel yönetimlere, özel sektöre, Üniversitelere, ve çeşitli kullanıcı gruplarına hizmet sunacaktır. UKVA'da yer alacak kamu kurumlarından bazıları Harita Genel Komutanlığı (HGK), İller Bankası, Karayolları, Tapu ve Kadastro Genel Müdürlüğü (TKGM), Devlet Su İşleri (DSİ), Devlet İstatistik Enstitüsü (DİE), Etibank, Köy Hizmetleri, Maden Tetkik Arama (MTA), Türkiye Petrolleri Anonim Ortaklığı (TPAO), Türkiye Elektrik Dağıtım A.Ş. (TEDAŞ), Arsa Ofisi Genel Müdürlüğü (AOGM), Orman Genel Müdürlüğü (OGM) dır. Bu kurumlar, kendi ürettikleri veriyi diğer kurumların kullanımı için UKVA'ya sunacaklar, ve kendi gereksinimleri olan veriyi yine UKVA'dan sağlayacaklardır. Örneğin Tapu Kadastro Müdürlükleri tapu bilgilerini UKVA üzerinden kullanıma sunacak, belediyeler imar uygulamaları için gerekli bu bilgileri UKVA'dan sağlayacaktır. Bu bakımdan UKVA'nın işlevi, ülke genelinde bir "Bilgi Pazarı" dır. Bunun için her bir kurumun UKVA'ya ne sağlayacağı ve UKVA'dan ne alacağı, veri tipleri bazında belirlenmelidir.

1.1.2.2 Konumsal Veri Değişim Standardı Problemi

UKVA'nın kurulması ve yaşatılması bir dizi gereksinimin karşılanmasını içerir. Söz konusu gereksinimler "politik", "yönetimsel", ve "teknik" olmak üzere üç sınıf altında düşünülebilir [18]. Politik gereksinimler, UKVA'nın kurulması ve yaşatılması ile ilgili temel politikaların belirlenmesine yöneliktir. Yönetimsel gereksinimler, UKVA'nın gerek hayata geçirilmesi ve gerekse yaşatılma sürecindeki yönetimi ile ilgilidir. Teknik gereksinimler,

sürekli gelişen teknolojinin çok yakından izlenerek UKVA gereksinimlerine uygun teknolojilerin seçimi ya da geliştirilmesi, seçilen teknolojinin en verimli bir biçimde kullanılması ile, gerektiğinde yeni teknolojilerin uyarlanmasına yöneliktir.

UKVA'nın en önemli teknik gereksinimlerinden biri, taraflar arasında etkin bir veri değişimine olanak tanıyacak bir konumsal veri değişimi standardıdır. UKVA için ayrıca, detay kodlama ve işaretleri standartları ile iletişim, yazılım, donanım standartları gereklidir. Bu tezin kapsamı, konumsal veri değişimi ile sınırlıdır. Bu nedenle, standartlar konusu burada yalnızca konumsal veri değişimi açısından ele alınacaktır. UKVA ile çok çeşitli kesimlerin, zaman içerisinde de değişebilecek çok farklı gereksinimlerinin karşılanması söz konusu olduğundan, UKVA için standart belirleme çözümü çok güç bir problemdir.

Yukarıda belirtildiği gibi UKVA, çeşitli kamu kurumları, özel sektör, yerel yönetimler gibi, çok çeşitli kesimlere hizmet sunacaktır. Bu kesimler ilgi alanları gereği karayolundan imar parseline, limandan su deposuna çok farklı veri türleri ile çalışırlar. Dahası aynı veri türleri, farklı kullanıcı grupları tarafından farklı şekillerde tanımlanır. Örneğin TCK ve HGK'nın karayolu tanımları birbirinden oldukça farklı olacaktır. Öte yandan, farklı koordinat, projeksiyon sistemleri, raster vektör gösterim, farklı işaretler gibi veri gösteriminden doğan farklılıklar söz konusudur. Bütün bunlara ayrıca, UKVA taraflarının kullandığı yazılım, donanım, ve iletişim ortamı farklılıklarını eklemek gerekir. Dolayısıyla, UKVA'nın bütün tarafları arasında eksiksiz veri değişimine olanak tanıyacak bir konumsal veri değişim standardının "genel", "esnek" olması gerekir. UKVA çok dinamik bir yapıdır; Zamanla yeni taraflar eklenebilecek ya da mevcut tarafların veri tanımlama ve gösterim gereksinimleri değişebilecektir. Bu nedenle standart aynı zamanda "genişletilebilir" olmalıdır. Ayrıca standardın iletişim ortamı ve donanımdan bağımsız olması, etkin bir veri değişimine olanak tanınması gerekir.

Anılan özellikleri taşıyacak bir konumsal veri değişim standardı için iki yol vardır. Birincisi standardın sıfırdan başlayarak geliştirilmesidir. Bu oldukça zor ve uzun vadeli bir iştir. A.B.D.'nin SDTS (Spatial Data Interchange Standard) standardının geliştirilmesi, geniş bir ekiple yaklaşık 9 yıl almıştır [22]. Üstelik sonuçta, geliştirilen ürünün "standart" olarak kabul görmesi, yalnızca teknik tasarıma bağlı değildir. Diğer bir ifadeyle teknik tasarım ne kadar iyi olursa olsun, bir takım teknik olmayan etkenlere formatın standartlaşmasını engelleyebilir [11]. Örneğin büyük bir organizasyon, kullanmakta olduğu bir değişim formatı yerine yeni birini uyarlamaya, teknik açıdan uygun olsa bile, sırf böyle bir geçişin getireceği külfet nedeniyle sıcak bakmayabilir. Ya da büyük bir KVYS yazılım şirketi, bir değişim standardını, yazılımlarına olan bağımlılığı azaltabileceği için desteklemeyebilir. Bu nedenle bugün herkes adeta "kendi standardını" kullanmaktadır.

Bununla birlikte teknik olmayan problemlerin giderilmesi, UKVA kapsamında daha kolay olabilir; Teknik olarak yeterliği kabul edilmiş bir UKVA standardının taraflarca kullanılması yasal zorunluluk olarak getirilebilir. Örneğin A.B.D.'de, Şubat 1994'den itibaren federal kuruluşlara kullandıkları diğer formatlar yanında, SDTS formatında da veri sağlamaları zorunluluğu getirilmiştir.

UKVA standardı için diğer seçenek, A.B.D. ve Kanada gibi CBS ler alanında öncü ülkelerde, özellikle son yıllarda geliştirilmiş ve standart olarak önerilmiş olan değişim formatlarından birinin Türkiye için uyarlanmasıdır. Genel ve aynı zamanda esnek bir konumsal veri değişim formatının oldukça kapsamlı ve karmaşık olacağı bir gerçektir. O nedenle böyle bir formatın değerlendirilmesi de kolay değildir. Uzun bir sürede, bir ekip çalışması sonunda geliştirilmiş olan bir standardın arkasındaki tasarım ilkelerinin çok iyi anlaşılması gerekir. Aksi durumda, "yanlış seçim" yapma olasılığı vardır.

1.1.3 Tezin Amacı ve Katkısı

Bu tezde amaç, Türkiye için oluşturulacak Ulusal Konumsal Veri Altyapısı (UKVA) gereksinimlerini karşılayacak ve UKVA'da etkin veri değişimine olanak tanıyacak, bir konumsal veri değişim standardının belirlenmesidir. Detay kodlama ve işaretoloji standartları bu tezin kapsamı dışındadır.

Yapılan çalışma sonunda, UKVA gereksinimlerini karşılayacak ve taraflar arasında etkin veri paylaşımına olanak tanıyacak bir veri değişim standardı önerilmiştir. Böylece, UKVA'da konumsal veri değişimi problemi yalnızca kısmen çözümlenmiş olmaktadır. Problemin tamamiyle çözümü için, seçilen standarda Türkiye'ye özel bir takım eklemeler yapılması, standart detay/öznitelik katalogları ve işaret kitaplıklarının geliştirilmesi gibi, daha bir dizi gereksinim karşılanmalıdır. Ancak bu tez ile, UKVA için ilk adım atılmış ve çok önemli bir gereksinim karşılanmış olmaktadır.

Tezin ayrıca teorik ve kavramsal katkıları vardır. Teorik katkısı, değişim formatlarının değerlendirilmesini bugüne kadar yapılmış benzeri çalışmalardan daha etraflı bir biçimde ele alarak, bir dizi değerlendirme ölçütü sunmasıdır. Bunlardan veri modeli ölçütleri, aynı zamanda bir CBS nin veri modelinin değerlendirilmesinde de kullanılabilir. Kavramsal açıdan tezin başlıca katkıları, 1.4 kısmında UKVA kavramının sunulması ve UKVA gereksinimlerinin belirlenmiş olması, 1.3 kısmında konumsal veri değişiminin literatürdekinden farklı, yeni bir yaklaşımla ele alınması, ve 1.2 kısmında konumsal veri modellerinin sınıflandırılması verilebilir. Tezde ayrıca, bir çok kavram yerli yerine oturtulmaya çalışılmıştır. Tezin diğer bir katkısı da terminoloji açısından, CBS ler alanında

tezde geçen İngilizce terimlerin Türkçe karşılıkları özenle belirlenmeye çalışılmıştır. Bunlar tezin ek kısmında yer almaktadır.

1.1.4 Yaklaşım ve Yapılan Çalışma

UKVA konumsal veri değişimi standardı için bu tezde benimsenen yaklaşım, A.B.D. ve Kanada gibi CBS ler alanında öncü ve aşama kaydetmiş ülkelerde, özellikle son yıllarda geliştirilmiş ve standart olarak önerilmiş olan değişim formatlarının belirli ölçütlere göre değerlendirilerek en uygun birinin seçilmesi ve Türkiye için uyarlanmasıdır. Bu, iki nedenden ötürü, Türkiye için yeni bir standardın geliştirilmesinden çok daha uygun olan yoldur. Birincisi, yeni bir standardın geliştirilmesi, bunun için gerekli emek ve zaman açısından akılcı değildir. Türkiye, diğer pek çok alanda olduğu gibi, CBS ler ve UKVA alanında da oldukça geri kalmıştır. Kanada ve A.B.D. gibi pek çok ülke, ulusal konumsal veri değişimi standardı geliştirme çalışmalarına ve veri paylaşımı programlarına 1970 ve 80'li yıllarda başlamışlardır. Gerek standart geliştirmede ve gerekse veri paylaşım programlarının pek çoğunda istenen başarı elde edilememiştir. Türkiye'nin tekrar böylesi deneyimlerden geçmesi gerekmez. Fakat Türkiye, söz konusu geri kalmışlığını bu deneyimlerden yararlanarak giderme şansına sahiptir. İkinci olarak, Türkiye'de kullanılan KVYS yazılımları büyük ölçüde standartları uyarlanması düşünülen ülkelerde kullanılan yazılımlardır. Değerlendirilecek değişim formatları zaten bu yazılımları kullanan sistemler arasında veri değişimine yöneliktir. Bu nedenle, bu formatların veri modelleri ortak özellikler taşır. Dolayısıyla, Türkiye için geliştirilecek yeni bir standart, veri modeli ve içermesi gereken diğer özellikler bakımından diğer ülke standartlarından çok farklı olmayacaktır.

Herhangi bir değerlendirmede olduğu gibi, konumsal veri değişim formatlarının değerlendirilebilmesi için de, bir takım değerlendirme ölçütlerine gereksinim vardır. Genel olarak, UKVA standardının genel, esnek, genişletilebilir, donanım ve iletişim ortamından bağımsız, gerçekleştirimi kolay ve etkin veri değişimine izin vermesi gerektiği yukarıda belirtilmişti. Bu özellikleri taşıyacak bir konumsal veri değişim formatının, oldukça kapsamlı ve karmaşık olacağı bir gerçektir. Öte yandan, konumsal veri değişim formatlarının değerlendirilmesi konusunda bugüne kadar yapılan çalışmalar gerek sayı ve gerekse kapsam olarak oldukça sınırlıdır. Bunlardan [20] ve [21]'in çalışmaları konuya çok genel olarak eğilmektedir. [12] ve [16]'ın çalışmaları daha kapsamlı olmakla birlikte, bir UKVA standardının değerlendirilmesi için yeterince ayrıntılı değildi. O nedenle bu tezde, anılan çalışmalardan da yararlanılarak bir dizi değerlendirme ölçütü belirlenmiştir. Bu ölçütler bir konumsal veri değişim formatını "veri modeli", "gerçekleştirim", ve "değişim ortamı" açısından değerlendiren ölçütlerdir.

Değerlendirme ölçütleri, pek çok geleneksel konumsal veri değişim formatı dokümanında ve yukarıda anılan çalışmalarda görülemeyen bir açıklıkla, veri modeli ve gerçekleştirim kavramlarını birbirinden ayırmaktadır. Veri modeli ile ilgili olarak onbir ayrı ölçüt belirlenmiştir. Bunlar veri modeli tipi, raster veri, kartoğrafik gösterim, koordinatlar, topoloji-geometri ayırımı, karmaşık nesne, nesne paylaşımı, meta veri, nesne bazında veri değişimi, zamansal veri, ve genişletilebilirlik ölçütleridir. Gerçekleştirim ve değişim ortamı için üçer ölçüt belirlenmiştir. Gerçekleştirim ölçütleri formatın özellikli transferler için sunduğu desteği, detay kodlama ve fiziksel kodlama yöntem ve standartlarını irdeleyen ölçütlerdir. Değişim ortamı ölçütleri ise, formata yönelik çevirici geliştirmede programcılara, ve veri değişimi için kullanıcılara sağlanan olanaklar ile formatın dokümantasyonuna yöneliktir. Daha genel olarak, değerlendirme ölçütleri ile aranan, veri modeli açısından genellik ve genişletilebilirlik, gerçekleştirim açısından esneklik ve donanım bağımsızlığı, ve değişim ortamı açısından çeviricilerin kolay ve hızlı bir şekilde geliştirilmesine ve etkin bir veri değişimine olanak tanıyacak servis ve araçlar ya da bunları içeren bir Grafik Kullanıcı Arabirimi (GKA) dir.

Değerlendirmeye alınacak formatların belirlenmesi için de bir "ön-eleme" yapılmıştır. Çünkü bugüne kadar, gerek teknik ve gerekse politik nedenlerle pek çok konumsal veri değişim formatı geliştirilmiştir. Bunlardan bazıları oldukça yaygın bir kullanım alanı bulmuş, hatta "defacto" standartlar olarak adlandırılmıştır. Bugün bir milyon dolayında kullanıcısı bulunan AutoCAD yazılımının veri değişim formatı olan DXF, defacto standartların en belirgin örneklerden biridir. Diğer bir örnek, A.B.D.'nin başlıca konumsal veri sağlayıcı kuruluşu olan, Amerikan Jeolojik Ölçmeler (USGS)'in sayısal veri dağıtımında bugüne kadar çok yaygın olarak kullandığı DLG formatıdır. Ancak, ön eleme sonucunda bu alışlagelmiş formatların UKVA'nın gerektirdiği genellik, esneklik, ve genişletilebilirlikten yoksun olduğu görülmüştür. Çünkü bu formatların çoğu, özellikli amaçlara yönelik olarak geliştirilmiş, oldukça dar kapsamlı formatlardır.

Ön eleme sonucunda, yalnızca üç format, değerlendirme ölçütlerine göre değerlendirmeye uygun bulunmuştur. Bunlar Kanada'da geliştirilmiş olan SAIF (Spatial Archive and Interchange Format), A.B.D.'nin SDTS (Spatial Data Transfer Standard), ve NATO'nun DGIWG (Digital Geographic Information Working Group) çalışma grubu tarafından geliştirilmiş olan DIGEST (Digital Geographic Information Exchange Standard) formatlarıdır. Bu formatlar yukarıda anılan alışlagelmiş formatlardan daha sonra geliştirilmiş formatlar olduklarından, yeni bir takım kavramlar içermektedir. Aynı zamanda bunlar, konumsal veri değişiminde gelecekte söz sahibi olacak formatlardır. Bu nedenle, söz konusu üç formatın birbirleriyle uyumluluğunun sağlanması çabaları son bir kaç yıldır hız kazanmıştır [23].

Değerlendirme sonucunda SAIF, UKVA gereksinimlerini karşılayacak ve etkin bir veri değişimine olanak tanıyacak en uygun format olarak bulunmuştur. Veri modeli açısından daha kapsamlı ve Nesneye-Dayalı veri modelini kullanması nedeniyle de daha genişletilebilir bir model sunması ile SAIF, SDTS ve DIGEST'den daha üstündür. SAIF'in yüksek bir genişletilebilirlik sunmasının, UKVA açısından önemi çok büyüktür. Çünkü UKVA'da çok farklı kullanıcı grupları olacak ve zamanla yenileri eklenebilecektir. Gerçekleştirim açısından SAIF, detay kodlama bazında yeni bir yaklaşım sunmakta ve SDTS ve DIGEST'in kullandığı alışlagelmiş detay kodlama yerine, farklı kullanıcılara detaylarını istedikleri gibi tanımlama olanağı sunmaktadır. Böylece, çok zor bir iş olan standart detay kodlama kataloglarının oluşturulması ve daha da önemlisi yaşatılmasından kaçınılmış olmaktadır. Değişim ortamı açısından da SAIF, çevirici geliştiriciler için DIGEST ve SDTS'e göre çok daha geniş olanaklar sunmaktadır.

1.1.5 Tez Konusu ile İlişkili Diğer Çalışmalar

Konumsal veri değişim standartlarının değerlendirilmesi konusundaki diğer çalışmaların yetersizlikleri yukarıda dile getirilmişti. Türkiye'de konumsal veri değişim standartlarının değerlendirilmesi ya da Ulusal bir konumsal veri değişim standardının belirlenmesine yönelik, yazarın bilgisi dahilinde olan bir çalışma yoktur. Yalnız, HGK'nın tez ile yakından ilişkili "Ulusal CBS projesi" vardır. Bu kısımda, tez konusu ile ilişkili ulusal, uluslararası, ve Türkiye'deki çalışmalar, özetle gözden geçirilmektedir.

1.1.5.1 Ulusal Çalışmalar

A.B.D. : CBS ler alanında öncü ve Dünya CBS pazarında en büyük pay sahibi CBS yazılımlarının geliştirildiği ülkelerden biri olması nedeniyle A.B.D., konumsal veri değişiminde standartlara hükmeder bir konumundadır. Konumsal veri değişiminde kullanılan DLG, DXF, SIF gibi, çok çeşitli formatların yerine, çok daha geniş kapsamlı bir standart geliştirme çalışmalarına 1982'de başlanmıştır. Bu çalışmalar sonunda geliştirilen SDTS formatı, 1992'de Ulusal standart olarak kabul edilmiş ve federal kurumlara, 1994 Şubat'ından itibaren SDTS formatında veri sağlama zorunluluğu getirilmiştir. A.B.D. son bir kaç yıldır UKVA çalışmalarına hız vermiştir. Bu amaçla USGS (US Geological Survey) liderliğinde, FGDC (Federal Geographic Data Committee) ve onun alt komiteleri ile, bir de MSC (Mapping Science Committee) oluşturulmuştur.

Kanada : Standart geliştirme çalışmaları CCOG (Canadian Council on Geomatics) ile 1970'lerde başlamış ve CCOGIF, MDIF, ve MACDIF gibi çeşitli formatlar geliştirilmiştir. British Columbia Eyaletince geliştirilen SAIF, 1991'de ulusal "genel" standart, DIGEST de

dar kapsamlı transfer standardı olarak kabul edilmiştir. Kanada da A.B.D. gibi, konumsal veri paylaşımı alanında çok büyük bir birikime sahiptir. Bugüne kadar pek çok veri paylaşım programı, değişen başarı derecelerinde uygulanmıştır. Halen başarı ile sürdürülen bu programlardan biri olan "LandData BC" ile, kullanıcılara BC eyaleti düzeyinde çeşitli arazi bilgisine (topoğrafya, kadastral durum vs.) çevrim-içi erişim olanağı tanınmaktadır [24].

Finlandiya : Konumsal veri paylaşımı alanında bugün oldukça ileri bir konumda bulunan Finlandiya'da çalışmalar 1970'lerde başlamıştır. 1985'te Tarım ve Orman bakanlığının öncülüğünde, farklı kurumlar arasında etkin veri paylaşımı için, "Coğrafi Bilgi Pazarı" (CBP) projesi başlatılmıştır. Ülke düzeyinde 20 kurum CBP ye veri sağlamak ve bu veriyi tanımlamakla görevlendirilmiştir. CBP'de veri değişimi bir ISO standardı olan EDIFACT (Electronic Data Interchange For Administration, Commerce and Transport) üzerinden sağlanır. Kullanıcılar bir sorgulama arabirimi ile, veri kaynağından bağımsız olarak standart bir sorgulama ile istedikleri veriye ulaşabilirler. CBP'nin diğer önemli bileşeni olan "coğrafi veri sözlüğü", hangi veri kaynağından ne tip ve hangi özelliklerde veri elde edilebileceğini gösterir [25].

Özetle diğer ülkelere de bakılacak olursa, İsveç parsel tapu ve kadastro bilgilerini içeren genel kullanıma açık bir veri tabanı oluşturma çabalarına 1970'lerde başlamış, bugün ülkenin büyük bir kısmı için, veri girişi tamamlanmıştır. Çeşitli özel ve kamu kesimleri, bu veri tabanına çeşitli iletişim ağları üzerinden günlük giriş yapmaktadır. İngiltere'de de standartlaştırma çabaları 1970'lere uzanmaktadır. 1985'ten itibaren NTF (National Transfer Format) geliştirilmeye başlanmış, 1989'da tamamlanmıştır. NTF genel bir standart olduğundan, formatın özellikli transferlerde kullanabilmesi için dört ayrı profili (alt formatı) tanımlanmıştır. Örneğin raster veri için "raster veri" profili kullanılır. Avusturalya ve Yeni Zelanda SDTS'i uyarlamaktadır. Japonya bu alanda oldukça geridir. Ancak 2010 yılında işlevsel hale getirmeyi planladığı Ulusal Bilgi Altyapısı (UBA) için 330 milyar Dolar ayırmıştır [26].

1.1.5.2 Uluslararası Çalışmalar

Bugüne kadar gerek teknik ve gerekse teknik olmayana nedenlerle çeşitli kesimlerin "kendi" standartlarını geliştirmiş olmaları sonucu, konumsal veri değişimi alanında bugün çok çeşitli "standartlar"ın kullanılıyor olması, standartlaştırma ile amaçlananın tam tersi bir durumdur. Kullanılan standart sayısının en aza indirilmesi amacıyla son bir kaç yıldır, DIGEST, SDTS, SAIF, ve DX-90 [27] gibi başlıca konumsal veri değişim formatları arasında uyum sağlanması çalışmaları gündeme gelmiştir [23]. Bu çerçevede, "genel" formatlar olan SDTS ve SAIF'in, DIGEST'e yönelik "profilleri" öngörülmüştür. Genel bir

standardın profili, standardın bir alt grubudur (vektör ve raster veriye yönelik bir formatın vektör profili gibi). SDTS'nin 1993'te DIGEST, 1994'te de DX-90 profilleri geliştirilmiştir. SAIF-DIGEST profili de 1993 te tanımlanmıştır. Böylece SAIF ve SDTS arasında uyum da, DIGEST üzerinden sağlanabilir. Ayrıca 1993/94'de DX-90 ve DIGEST arasında eksiksiz bir çeviri için, her iki formata bir takım değişiklikler yapılmış, FACC, IHO detay tanımlarını da içermek üzere genişletilmiştir. Öte yandan FACC, çeşitli ülkelerin gereksinimleri doğrultusunda genişletilmekte ve DIGEST'den bağımsız, Dünya çapında bir katalog haline getirilmesi düşünülmektedir [23].

Konumsal veri yönetiminde Avrupa ülkeleri arasında standardın sağlanması amacıyla, Avrupa Standartlar Organizasyonu (CEN-European Committee for Standardization) tüm üye ülkelerin coğrafi bilgi için kullanacakları standartları belirlemek üzere, 1991'de "TC 287" teknik komitesini kurmuştur. TC 287 coğrafi bilgi için, "geometri", "zamansal veri", "meta veri" gibi çeşitli standart düzeyleri belirlemiştir. Her bir düzeye ait standartlar ile, bir konumsal veri değişim standardının ilk versiyonları, 1995 sonlarında çıkmış ve halen TC 287'nin incelemesindedir [28]. TC 287, konumsal veri değişimini, bilinen değişim formatlarından çok daha geniş bir kapsamda ele almakta ve bir "referans modeli" tanımlamaktadır [29]. Buna göre, konumsal veri değişiminin üç düzeyi vardır; "Veri tanımlama", "servis ve protokol", ve "veri iletişimi". Veri tanımlama, soyut ve somut tanıma, servis ve protokol çevirilere, veri iletişimi de verinin bir taraftan diğerine iletişimine işaret eder. Referans modeli, veri tanımlama için grafik ve metinsel veri tanımlama dilleri, servis ve protokol düzeyinde "değişim", ve veri iletişimi düzeyi için de "iletişim" servisleri tanımlamaktadır. TC 287, veri değişimi için bir de "değişim mimarisi" tanımlamaktadır. Değişim mimarisinde iki sistem arasında veri değişimi, verinin sunucu ve alıcı tarafındaki tanımlamaları üzerinden, değişim ve iletişim servisleri yardımıyla gerçekleştirilir [29]. Bu servisler için gerekli destekleyici yazılımların bir kısmı mevcut, diğer bir kısmı da geliştirilmektedir [28].

CEN TC 287 gibi, konumsal veri değişimini "interoperabilite" kapsamında ele alan diğer bir organizasyon, bir OGIS (Open Geodata Interoperability Specification) geliştirmek üzere, 1993 yılında A.B.D.'de oluşturulan OGC (Open GIS Consortium) dir. Farklı yazılım, donanım, ve veri modellerine sahip CBS lerden oluşan "dağıtılmış bir ortamda" kullanıcıların uygulamalarını, herhangi bir sistemdeki veri ya da yazılımları, hangi sistemde bulunduklarından bağımsız ve doğrudan kendi uygulamaları içerisinden, kullanarak gerçekleştirebilmeleri "interoperabilite" olarak anılmaktadır. OGC 40'ın üstünde kamu ve özel sektör organizasyonu tarafından desteklenmektedir. OGIS için oluşturulan teknik komite üyeleri ise, başlıca KVYS ve donanım firmaları, veri sağlayıcıları, ve son kullanıcılarıdır [30].

OGIS'in iki temel bileşeni vardır. Bunlar OGM (OGIS Geodata Model) ve ORM (OGIS Reference Model) dir [31]. OGM, genel ve genişletilebilir bir veri modeli sağlayacak ve aynı zamanda SDTS, SAIF, ve DIGEST formatları üzerine bir arabirim tanımlayacaktır. OGM, veri tipleri yanında bu veri tipleri üzerine uygulanabilecek işlemleri ("servisler", örn. "poligon" veri tipi için "alan-hesabı") de tanımlayacaktır. ORM ise, dağıtılmış bir ortamda bu servislerin gerçekleştirilmesinin istenmesine yönelik standart yöntemleri belirleyecektir. OGIS Mimarisi'nde, SQL (structured Query Language)'in konumsal veri ve multimedya için genişletilmiş, Nesneye-Dayalı uyarlaması olan SQL3-MM kullanılacaktır. Böylece, bir kullanıcı kendi uygulaması içerisinde, diğer sistemlerdeki veriyi sorgulayabilecek, bu veri üzerinde birtakım işlemler gerçekleştirebilecektir. Burada kullanıcının kendi uygulamasından çıkmaması, bir Microsoft Word kullanıcısının Word içerisinde Spreadsheet analizlerini gerçekleştirmesine benzetilebilir. Dağıtılmış bir ortamda bir kullanıcının bulunduğu sistemden, diğer bir sistemdeki veri üzerinde analizler yapılabilmesi ile, veri değişimi gerekmeyebilecektir [31]. Bu tabi ulaşılması çok kolay bir hedef değildir.

OGIS ortaya atıldığından beri, CBS endüstrisini önemli ölçüde etkilemiş, aynı zamanda OGC üyesi olan hemen bütün büyük CBS firmaları, sistemlerini daha "açık" yapma yarışına girmiştir. OGIS aynı zamanda, A.B.D.'de UKVA açısından büyük önem taşımaktadır. Çünkü OGIS ile UKVA'nın teknik altyapısı oluşturulmuş olacaktır. OGIS kapsamının çok geniş ve OGC'nin görüş birliğine dayalı bir organizasyon olması, ilk versiyonun ne zaman tamamlanabileceği konusundaki tahminleri güçleştirmektedir [32].

Öte yandan, Uluslararası standartlar organizasyonu (ISO-International Standards Organization) 1994'de çeşitli ulusal ve uluslararası standartlaştırma çabalarını tek bir şemsiye altında toplamak amacıyla "TC 211" teknik komitesini oluşturmuştur. ISO TC 211 kapsamında birleştirilmesi düşünülen çalışmalar IHO (International Hydrographic Organization), NATO'nun DGIWG (Digital Geographic Information Working Group) grubu, ve CEN TC 287'nin çalışmalarıdır. Ayrıca SDTS, SAIF, ve NTF formatlarından, ISO TC 211 kapsamında yararlanılması düşünülmektedir. Çeşitli ülkeler, ISO TC 211'i katılımcı ya da gözlemci statüsü ile desteklemektedir. Türkiye, Türk Standartlar Enstitüsü (TSE) ile katılımcı üyedir. ISO TC 211, coğrafi veri için "GOSE (Geographic Open Systems Environment)" ortamını tanımlamakta ve böyle bir ortamın gerçekleştirilmesinde konumsal veri yönetim ve işleme standartlarının kullanılmasını öngörmektedir [33]. ISO TC 211'in çalışmalarının ne zaman tamamlanabileceği belli değildir.

1.1.5.3 Türkiye'deki Çalışmalar

HGK, gerek Türk Silahlı Kuvvetleri ve gerekse diğer kamu kurumlarının 1:25000 ve daha küçük ölçekli harita bilgisi gereksinimlerini karşılamak üzere Haziran 1990'da "Ulusal CBS projesi" ni başlatmıştır [34]. Bu proje ile amaçlanan, anılan bilgi gereksinimini çevrim-içi tarzda karşılayacak bir Coğrafi Veri Tabanı (CVT)'nin oluşturulması idi. Bu amaçla 19 proje aşaması belirlenmiş, 14. aşamada "Ulusal coğrafi veri değişimi transfer formatı tasarlanması" öngörülmüştür. Ancak bu aşamaya henüz gelinmemiştir. Projenin tamamlanan 6. aşamasında, 1:25000 ve 1:10000 ölçekli harita detayları için, bir detay/öznitelik katalogu hazırlanmıştır. Bunun için FACC'tan yararlanılmıştır. Bu çalışma UKVA için önemli bir katkı sağlayacaktır.

Öte yandan, HGK'nın bu çalışmasının diğer kamu kurumlarının aktif katılımıyla genişletilmesi amacıyla, Devlet Planlama Teşkilatı (DPT) önderliğinde DİE, daha geniş bir detay/öznitelik katalogu oluşturmakla görevlendirilmiştir. Ancak bütün bu çalışmaların, kurumlararası görüş birliğinin sağlanmasında karşılaşılan güçlük nedeniyle oldukça yavaş gittiği gözlenmektedir. O nedenle yapılması gereken, UKVA için gerekli adımın bir an önce atılması ve söz konusu çalışmaların, UKVA kapsamında ve bu tezde elde edilen sonuçlar ışığında yeniden ele alınmasıdır.

1.1.6 Tezin Düzenlemesi

Tezin bu bölümü, dört kısım olarak düzenlenmiştir. Giriş kısmının ardından 2. kısımda, veri modeli kavramları tanıtılmaktadır. Konumsal veri değişimi, veri modelleri ile doğrudan ilişkili olduğu için, bu kısımda konumsal veri için kullanılan veri modelleri ele alınmaktadır. 3. kısımda konumsal veri değişimi, yeni bir bakış açısıyla ele alınmaktadır. Bu kısımda ayrıca, Nesneye-Dayalı bir konumsal veri değişim ortamı tanıtılmaktadır. 4. kısım, UKVA kavramının açıklanmasına ayrılmıştır. Bu kısımda ayrıca, UKVA'nın oluşturulması için yapılması gerekenler genel olarak belirtilmiştir. Bu çalışmada belirlenen konumsal veri değişimi formatı değerlendirme ölçütleri, 2. bölümde ayrıntılı olarak tanıtılmaktadır. 3. bölümde, değerlendirmeye alınan değişim formatlarının, değerlendirme ölçütlerine göre incelenmesinden elde edilen bulgular yer almaktadır. Bu bulguların irdelenmesi ve nihai değerlendirme, 4. bölümde yapılmakta ve UKVA için bir konumsal veri değişim standardı önerilmektedir. Bu bölümde ayrıca, belirlenen standarda Türkiye için yapılması gereken eklemeler ile UKVA'da konumsal veri değişimine yönelik diğer gereksinimlere değinilmektedir. Sonuç ve öneriler sırasıyla, 5 ve 3. Bölümlerde verilmektedir. Tezin ek kısmında, İngilizce terimler ve çoğu bu tezde belirlenen Türkçe karşılıkları yer almaktadır.

1.2. Konumsal Veri modelleri ve Veri Yapıları

Konumsal veri değişimi, aynı bir gerçeğin farklı sayısal gösterimleri arasında bir çeviri içerir "Gerçek" (Reality, Real World, Miniworld, Phenomena) sözcüğüyle, sayısal gösterime konu olan gerçek dünya kastedilmektedir. Gerçeğin sayısal gösterimini oluşturmadaki farklı amaçlar, gerçeğin farklı şekillerde algılanmasına neden olur. Sözgelimi, bir orman mühendisi ile bir mimar, harita üzerinde farklı bilgiler görmek isterler. Gerçeğin belli bir perspektiften algılanarak ya da soyutlanarak, sayısal olarak gösterilmesinde veri modelleri ve veri yapıları kullanılır. Bu bakımdan, konumsal veri değişimi veri modelleri ve veri yapıları ile çok yakından ilişkilidir. O nedenle, tezin bu kısmı sözkonusu kavramların açıklanmasına ayrılmıştır. Ancak veri modelleri ve veri yapıları çok geniş konular olduğundan, bu kısımda yalnızca tezin diğer kısımlarının daha iyi anlaşılmasına yardım edecek ayrıntıda bilgi verilmektedir. Daha ayrıntılı bilgi için, bu kısımda anılan kaynaklara başvurulabilir.

Bu kısım ayrıca, bazı kavram karışıklıklarına parmak basması ile dolaylı bir katkıda bulunmaktadır. Örneğin, veri modelleri ve veri yapıları kavramları sıkça karıştırılmakta ve biri diğerinin yerine kullanılmaktadır. "Veri modeli" sözcüğü, çok yaygın olarak karşılaşılan diğer bir kavram karışıklığına yol açmaktadır: "Veri modeli" iki farklı anlamda kullanılmaktadır. Birinci kullanımda, gerçeğin belirli bir perspektifte soyutlanmış şekli veri modeli olarak anılmaktadır. Buna paralel olarak, gerçeğin belirli bir perspektifte soyutlanması da bazen "veri modelleme" olarak adlandırılır. Son zamanlarda daha yaygın olarak tercih edilen kullanıma göre ise, veri modeli gerçeğin bir soyutlamasını tek anlamlı olarak tanımlamayı sağlayan kavramlar ve kurallar bütünüdür. Gerçeğin belli bir perspektifte soyutlanmış halinin onun bir modeli olduğu, genel olarak doğru olmakla birlikte, birinci kullanım yeterince duyarlı değildir. Çünkü aşağıda açıklanacağı üzere, aynı veri modeli ile aynı bir gerçek için farklı soyutlamalar (tanımlamalar) elde edilebilir.

Konumsal veri değişiminde, transfer veri grubunda yer alan verinin anlamı kavramsal şema, sözdizimi (formatı) ise kavramsal ve fiziksel şema ile tanımlıdır. Diğer bir anlatımla, transfer veri grubundaki veri anlam ve format olarak hem kavramsal ve hem de fiziksel veri modeline bağlıdır. O nedenle bu kısımda konumsal veriye yönelik veri modelleri hakkında bilgi verilecektir. Bu kısım üç alt kısım olarak düşünülmüş, birinci alt kısım veri modeli kavramını açıklanmasına ayrılmıştır. İkinci alt kısımda, konumsal veri modelleri ve kullandıkları genel amaçlı veri modelleri tanıtılmaktadır. Üçüncü alt kısımda ise, konumsal veriye yönelik fiziksel veri modelleri ele alınmaktadır.

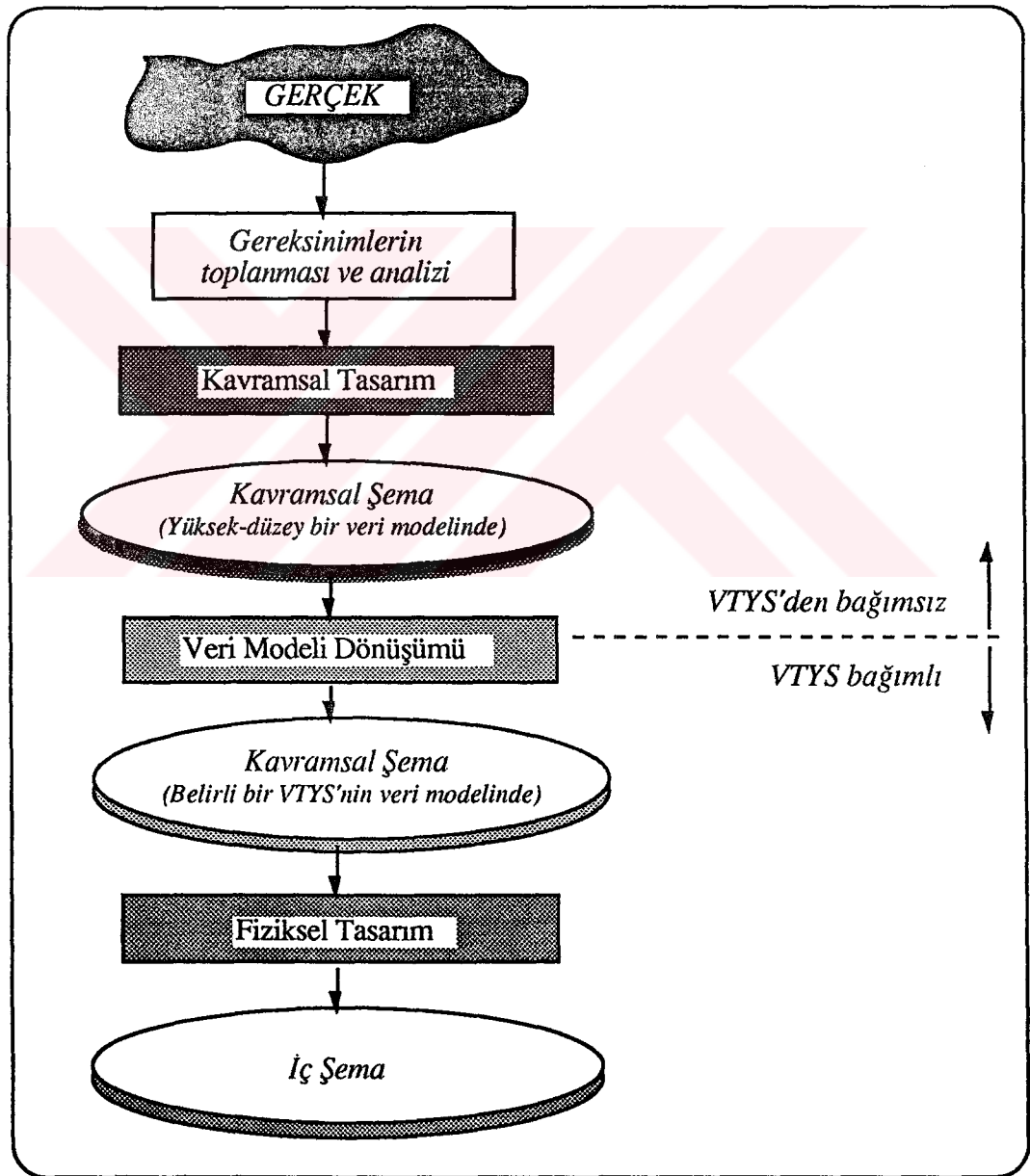
1.2.1 Veri Modeli Kavramı

Veri modeli için literatürde [29], [35] çeşitli tanımlar bulunmakla birlikte genel olarak, "veri tabanı yapısını tanımlamada kullanılan kavramlar bütünü" olarak tanımlanabilir [36]. Bir veri tabanı tasarımında, veri tabanının konusunu oluşturan "gerçek", bir veri modelinin sağladığı kavramlar yardımıyla belli bir perspektifte algılanarak (soyutlanarak), tek anlamlı olarak tanımlanır. Bu tanımlamanın iki düzeyi vardır. Yüksek düzey "kavramsal" düzeydir ve bu düzeydeki bir tanımlama "kavramsal şema" olarak anılır. Kavramsal şema kullanıcıların veri tabanı yapısını anlamalarına ve böylece uygulamalarını modellemelerine hizmet eder. Kavramsal düzey için kavramsal ya da gerçekleştirim (mantıksal) veri modelleri kullanılır. Düşük düzey ise, fiziksel düzeydir. Fiziksel düzeydeki bir tanımlama, fiziksel ya da daha yaygın olarak "iç şema" olarak anılır. İç şema, Veri Tabanı Yönetim Sisteminin (VTYS) anlayacağı dilde veri tabanı tanımını sağlar, fiziksel depolama yapıları ve veri erişimine ait detaylar içerir.

Veri tabanı tasarımı, veri tabanının konusunu oluşturan gerçeğin, veri tabanının oluşturuluş gereksinim ve beklentileri çerçevesinde soyutlanarak veri tabanına aktarılmasını içerir. Oluşturulacak bir veri tabanı farklı kullanıcı gereksinimlerini karşılayabilmelidir. Örneğin bir seyahat acentesi sekreteri için, verilen bir tarih ve iki merkez arasında, hangi havayollarından kaç, kaç bilet bulunabileceği türü sorular önemli iken, bir harita mühendisi için, bir yolun iki yanında, yola belirli uzaklıktaki kadastro parsellerinin bilgisayar ekranından görüntülenebilmesi veya bu parseller hakkında bilgi edinilebilmesi "olmazsa olmaz" türü bir gereksinim oluşturabilir. O nedenle veri tabanı tasarımında ilk adım, olası veri tabanı kullanıcı gereksinimlerinin belirlenmesidir. Söz konusu gereksinimler, veri tabanında yer alacak veri tiplerini ve verinin fiziksel olarak depolanması için kullanılacak olan veri yapılarını belirler.

Gerçeğin veri tabanındaki sayısal gösterimi, onun belli bir perspektiften bir modeli olup, bir VTYS'de gerek kullanıcılar ve gerekse bilgisayar tarafından anlaşılabilir bir tarzda tanımlanması gerekir. Böyle bir tanımlama, veri tabanı literatüründe "şema" olarak adlandırılır. Veri tabanındaki veri, sürekli olarak işleme tabi olduğundan değişkendir. Yeni kayıtlar girilirken, eski kayıtlar silinebilir ve hatta mevcut kayıtlar değiştirilebilir. Belirli bir andaki veri tabanı, bir "veri tabanı örneği" dir ve belirli bir şemaya uyar [36]. Dolayısıyla, verilen bir şema için sonsuz veri tabanı örneğinden söz etmek mümkündür. Çok kaba olarak şema, bir programda bir değişkenin "tamsayı" tipinde tanımlanmasına, veri tabanı örneği ise programın koşum anında bu değişkene değişik tamsayı değerler atanmasına benzetilebilir.

Geleneksel veri tabanı tasarımı, kavramsal düzeyden fiziksel düzeye doğrudur [36] (Şekil 2). Kavramsal tasarımda, gereksinimlere göre kavramsal şema belirlenir. Kavramsal şema tanımlamada, "kavramsal" ya da "mantıksal" veri modelleri kullanılabilir. "Kavramsal şema", ortalama veri tabanı kullanıcısı için, veri tabanının yapısını genel olarak tanımlar. Kullanıcıların veri tabanının yapısını anlamalarına ve böylece uygulamalarını modellemelerini sağlar. Kavramsal şema, fiziksel depolama yapılarının ayrıntılarına girmeden, varlıklar, veri tipleri, varlıklar arasındaki ilişki tipleri, ve kısıtlayıcılar üzerinde yoğunlaşır. Bu bakımdan kavramsal şema, yüksek düzeyli bir tanımlamadır. Diğer bir ifadeyle, kavramsal şema yazılım ve donanımdan bağımsızdır [37], [38].



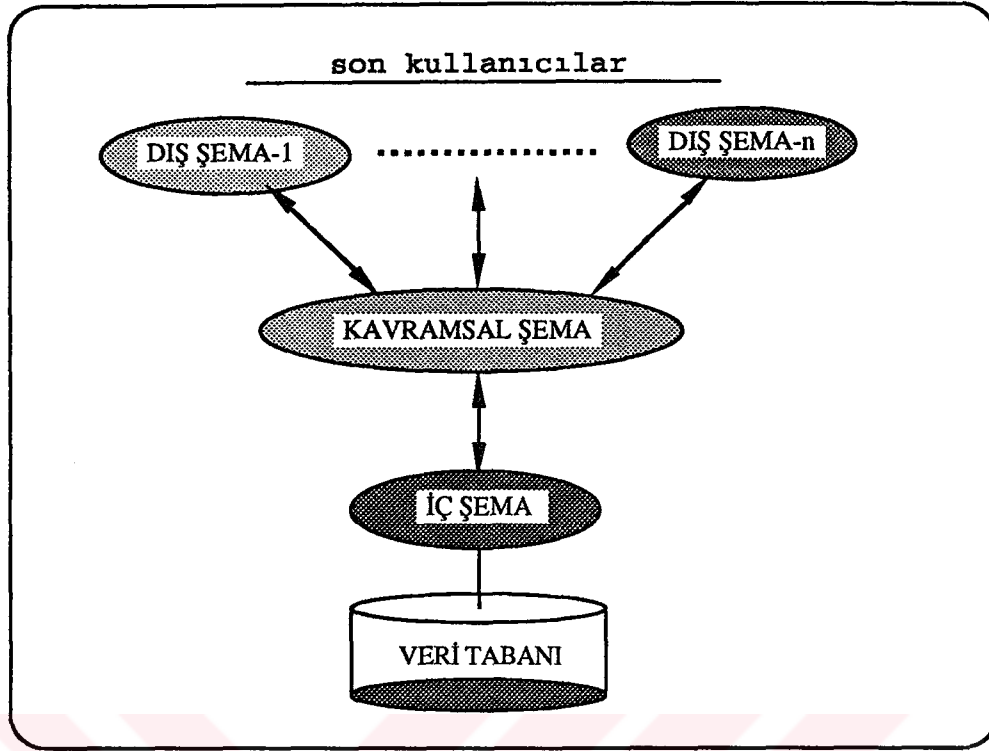
Şekil 2. Veri tabanı tasarımı.

Kavramsal veri modelleri oldukça yüksek düzeyli olduklarından, kavramsal bir veri modelinde tanımlı bir şema genellikle doğrudan gerçekleştirilemez. Bu nedenle, geleneksel veri tabanı tasarımında kavramsal tasarımdan sonraki adım çoğunlukla, gerçekleştirim için kullanılacak bir VTYS'nin seçimidir. Öte yandan, bugün piyasadaki VTYS'lerin çoğu mantıksal bir veri modeli kullanmaktadırlar. O nedenle mantıksal veri modelleri, "gerçekleştirim veri modelleri" olarak da bilinirler. O halde, kavramsal veri modelindeki kavramsal şema, VTYS'nin veri modelinde yeniden tanımlanmalıdır. Buradaki işlem, iki veri modeli arasında bir dönüşüm olup, bazen "mantıksal veri tabanı tasarımı" olarak anılır [39].

Fiziksel tasarım safhasında, verinin en yüksek verim için, veri tabanında fiziksel olarak nasıl organize edilmesi gerektiği belirlenir. Sonuç, iç şema dır. İç şema depolama yapılarını, kayıt formatlarını, kayıt alanlarını, veri tabanına giriş yol ve yöntemleri ile veri tabanının fiziksel gerçekleştirimini ilgilendiren diğer bütün detayları tanımlar. İç şema tanımlamada, genellikle "veri yapıları" olarak anılan, "fiziksel veri modelleri" kullanılır. İç şema yazılım ve donanım bağımlıdır [37], [38]. Fiziksel veri modelleri 1.2.3 te ele alınmaktadır.

Aynı bir veri modeli kullanılarak, aynı bir gerçeğin farklı şemalarını elde etmek mümkündür. Aslında bu, farklı kullanıcıların aynı bir gerçeği değişik şekilde algılamalarının doğal bir sonucudur. Örneğin, bir harita mühendisi ile bir kent plancısı aynı bir kent manzarasını farklı şekillerde algılayıp modelleyeceklerdir. Çünkü ikisinin de ilgi alanları ve dolayısıyla gereksinim duyduğu bilgiler farklıdır. Bu bakımdan, belirli bir kullanıcı grubu, yalnızca kendi uygulamalarını ilgilendiren, yani gerçeği kendi penceresinden tanımlayan "dış şema" lar oluşturabilir (Şekil 3). Her bir dış şema, veri tabanını belirli bir kullanıcı grubunun gereksinim duyduğu biçimde, ilgisiz detayları adeta gizleyerek tanımlar [36]. Dış şema mekanizması, değişik kullanıcı uygulamalarının modellenmesinde, büyük bir esneklik getirmektedir. Bugün piyasadaki pek çok CBS dış şema desteği sağlamaktadır [40]. Dış şema tanımlamada kavramsal ve gerçekleştirim veri modelleri kullanılabilir. Dış şema tanımlamada kullanılan veri modeli, kavramsal şemanın tanımlandığı veri modeliyle aynı ya da farklı olabilir.

Bir VTS'deki bu dış şema, kavramsal şema, ve iç şema yapısı, Amerikan Ulusal Standartları Enstitüsü (ANSI)'nin "Üç şema mimarisi" olarak bilinir [41]. Buna göre, dış şema ile kavramsal ve kavramsal şema ile iç şema arasında sürekli, bir dönüşüm vardır (Şekil 3). Dış şema tanımlamada kullanılan veri modelinin, kavramsal şemanın tanımlandığı veri modelinden farklı olması durumunda, dış şemadan kavramsal şemaya geçiş, sözkonusu veri modellerinin aynı olması durumuna göre daha zor olacaktır.



Şekil 3. Üç-şema mimarisi.

Üç-şema mimarisine göre, veri tabanı şemaları birbirinden bağımsız olmakla birlikte, "veri sözlüğü" üzerinden birbirleriyle ilişkilendirilmişlerdir. Veri tabanı şemaları arasındaki dönüşümlere ait kurallar yanında, veri tabanı şema tanımları, çeşitli VTYS yazılımlarının gereksinim duyduğu bilgilerle, tasarım kararları, ve uygulama programlarına ait bilgiler veri sözlüğünde bulunur [36], [40]. Böylece gerek kullanıcı programları ve gerekse VTYS yazılımları, veri tabanını hakkında gereksinim duydukları bilgiyi veri sözlüğünden alırlar.

1.2.2 Konumsal Veri Modelleri

Bir veri modelinin anlatım gücü, içerdiği kavramlarla sınırlıdır. Anlatım gücü ne kadar yüksekse, gerçek olduğuna (algılandığına) o derece yakın bir biçimde tanımlanabilecektir. Örneğin veri modeli eğer doğrudan, "A parseli B'nin KOMŞU parselidir" deme olanağını tanımıyorsa, yani "komşuluk" kavramını içermiyorsa, o zaman bu gerçek ancak dolaylı yoldan gösterim edilebilecektir. Bununla birlikte, veri modelinin kavramsal zenginliği ile kullanım kolaylığı iyi dengelenmelidir [39]. Diğer bir anlatımla, veri modeli doğrudan "gerçekleştirilebilir" olmalıdır. Anlamca zengin fakat oldukça karmaşık bir modelin doğrudan gerçekleştirimi çok güç olabilir. Bu durumda gerçekleştirimin, kavramsal olarak daha az zengin olan bir gerçekleştirim veri modeli üzerinden olması daha pratik olabilir ki, bu da kavramsal düzeyden gerçekleştirime geçişte "anlam" kayıplarına (örn. komşuluk ilişkisinin doğrudan belirtilememesi) neden olacaktır.

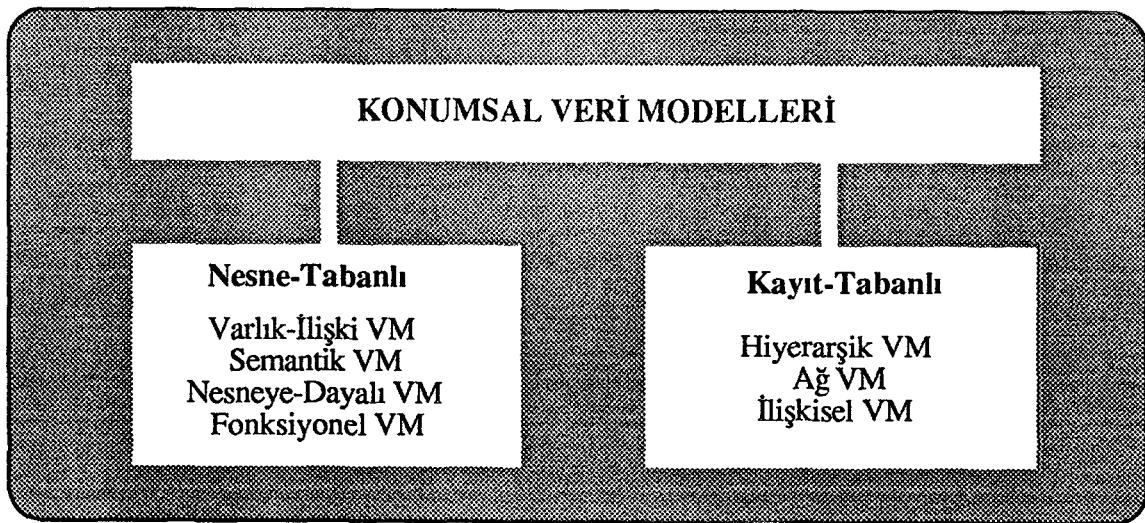
Bu tezde "konumsal veri" ile, Konumsal Veri Yönetim Sistemlerinin (KVYS) ilgi alanını oluşturan, konuma dayalı veri kastedilmektedir. Buna göre, kendi başına konumsal olan (örneğin bir karayolunun konumu) veri yanında, bir konum verilebilen fakat kendi başına konumsal olmayan (örn. karayolunun genişliği, yapım yılı, kaplama türü vs.) veri de "konumsal veri" kapsamındadır. Bu tür veriyle uğraşan sistemlerden bazıları BDT/BDÜ (Bilgisayar Destekli Tasarım/Üretim) sistemleri, CBS (Coğrafi Bilgi Sistemleri), GİS (Görüntü İşleme Sistemleri), BTS (Bilgi Tabanlı Sistemler) dir. Bu sistemlere konu olan uygulamalar ise, bir deprem bölgesinde bir anayola örneğin 10 km uzaklıktaki yangın istasyonlarının saptanması, veya hastanelerin fay hattına yakın olup olmadığının araştırılması, yeni yapılacak bir yol için en uygun güzergahın ve varsa kamulaştırılacak parsellerin ve değerlerinin tesbiti, Uydu görüntülerinin "sınıflandırılarak" belli bir bölgenin yeraltı ve yerüstü kaynaklarının saptanması, savaş durumunda örneğin helikopter iniş ve erzak bırakma yerlerinin saptanması türü uygulamalardır.

Geleneksel veri modelleri konumsal veri türünden, karmaşık veri ile başa çıkmada yetersiz kalmaktadır. Bunun nedeni bu modellerin esasen karmaşık olmayan veriye yönelik olarak geliştirilmiş, genel amaçlı modeller olmalarıdır. Bu yüzden geleneksel veri modelleri, karmaşık verinin olduğuna—ya da algılandığına—en yakın bir biçimde tanımlanması ve gösterim edilmesi için gerekli kavramsal zenginliğe ve gerçekleştirim desteğine sahip değildir. "Karmaşık olmayan veri" ile genellikle ticari uygulamalara konu olan veri kastedilir. Bu tür veri, "basit" olarak nitelenen kişi, iş, proje, havayolu şirketi, yayım gibi varlıklardan oluşmaktadır. Basit varlıklar sınırlı sayıda özelliklere sahiptir ve aralarında karmaşık ilişkiler bulunmaz. Bankacılık sistemleri, uçak rezervasyon sistemleri, kitaplık sistemleri bu tür veri kullanırlar. Bu sistemler için tipik uygulamalar örneğin bir banka sisteminde hesap açma kapama, mudi izleme, uçak rezervasyon sistemlerinde yer ayırtma, verilen bir tarih için yer arama vb. gibi işlemlerdir.

"Konumsal veri", hem veriyi oluşturan varlıkların kendi iç yapıları ve hem de bu varlıklar arasındaki ilişkiler bakımından "karmaşık"tır. Örneğin bir "drenaj ağı" akarsular, göller ve göller içerisinde adalardan oluşabilir. Biraz daha ileri gidilerek, bir dizi drenaj ağının daha büyük bir drenaj ağı oluşturmak üzere bir araya getirildiği düşünülürse, bir konumsal nesnenin belirli bir hiyerarşi içinde iç içe nesnelerden oluşabileceği anlaşılabılır. Geleneksel veri modelleri böyle bir varlığın olduğuna en yakın bir biçimde tanımlanabilmesi için, yeterli kavramsal zenginliğe sahip değildir. Örneğin akarsu ve göller arasındaki bağlantıları tanımlayacak bir "BAĞLI", ya da bir adanın hangi göle ait olduğunu göstermek üzere bir "SINIRLI" kavramları (ilişki tipleri) yoktur.

Gerçekleştirim açısından bakıldığında, geleneksel veri modelleri "kayıt-tabanlı" modellerdir. Diğer bir anlatımla, gerçekleştirim için verinin uygun tipte kayıtlara atanması gerekmektedir. Ancak yine basit veriye yönelik olarak geliştirilmiş olmalarının bir sonucu olarak, bu modeller için öngörülmesi olan kayıtlar, oldukça basit veri tiplerinden oluşmaktadır. Oysa konumsal veri, karmaşık veri tipleri ve karmaşık ilişkiler içermektedir. Bu durumda karmaşık verinin basit yapıları kayıtlar üzerine aktarılmasında sorunlar yaşanmakta, "anlam kayıpları" ortaya çıkmakta ve gerçek ancak bir takım anlamsal kayıplarla veri tabanında gösterim edilebilmektedir. Bir sorun, değişen uzunluktaki (x, y olarak) çizgilerin, sabit-uzunluklu kayıtlara aktarılmasıdır. Anlamsal kayıplar açısından, yitirilen anlamın uygulama programlarında, bir şekilde yeniden kazanılması gerektirmekte, bu da uygulama programlarının ve programcının yükünü çok önemli ölçüde artırmakta ve verimsizliğe yol açmaktadır. Örnek olarak, "SINIRLI" ilişki tipinin veri modelinde tanımlı olmaması nedeniyle, bütün adaları ait oldukları göllerle listelemek isteyen bir programcı bunun için özel bir program geliştirmek durumundadır.

Çözüm, veri modellerinin anlatım güçlerini artırmak ve böylece gerçeğin algılandığına en yakın bir biçimde gösterim edilebilmesini sağlamaktır. Bu amaçla konumsal veri için bugüne kadar çok çeşitli "konumsal veri modelleri" geliştirilmiştir. Konumsal veri modelleri genel amaçlı veri modelleri üzerine yeni bir takım kavramlar ekleyerek onları anlamca daha zengin bir duruma getirirler. Örnek olarak, İlişkisel modeli kullanan bir konumsal veri modelinin "SINIRLI" kavramını eklemesi verilebilir. Konumsal veri modelleri kullandıkları genel amaçlı veri modellerine bağlı olarak, "kayıt-tabanlı" ve "nesne-tabanlı" olmak üzere iki ayrı grupta düşünülebilir (Şekil 4).



Şekil 4. Konumsal veri modellerinin, kullanabilecekleri genel amaçlı veri modellerine göre bir sınıflaması.

Katman-tabanlı ve nesne-tabanlı modeller arasında birinden diğerine geçiş mümkündür [15], [16]. Aşağıda, kayıt-tabanlı ve nesne-tabanlı konumsal veri modelleri ve bunların kullandıkları genel amaçlı veri modelleri hakkında özet bilgi verilmektedir.

1.2.2.1 Kayıt-Tabanlı Konumsal Veri Modelleri

Kayıt-tabanlı modellerde gerçek, sıfır, bir, ve iki boyutlu temel geometrik elemanlar—nokta, çizgi, poligon—yardımıyla gösterim edilir. Bu geometrik elemanların birbirleriyle olan konumsal ilişkileri (komşuluk, bağlılık gibi) "topolojik" olarak tanımlanır ve bu ilişkiler topolojik veri yapıları kullanılarak gösterim edilir [42]. Geometrik elemanlar veri tabanında, gerçekte neyi gösterim ettiklerine dair öznitelik bilgileri ile ilişkilendirilirler. Örneğin bir çizgi bir otoyolu gösterim edebilir. Gerçekleştirmede veri tabanı, genellikle öznitelik bilgilerine göre düzenlenmiş "katman" lar halinde organize edilir. Örneğin, sadece karayollarını içeren bir katman gibi. Bu tip modeller için en çok kullanılan diğer isimler "katman-tabanlı" [43] ve "geometri-tabanlı" [16] dır.

Her ne kadar çoğu alışlagelmiş CBS [42], [44] başlangıçta belirli bir veri modeli esas alınarak geliştirilmemiş ise de, bu sistemlerin kayıt-tabanlı bir veri modeline sahip olduğu söylenebilir. Bu sistemler genellikle ilişkisel veri modelini kullanırlar. Öznitelik verisi, bir İVTYS yardımıyla çekip çevrilirken, konuma dayalı veri özel veri yapıları kullanılarak organize edilir. Konuma dayalı veri ile öznitelik verisi arasındaki ilişki "anahtar" lar yardımıyla sağlanır (Örn. bir parselin konumu ile, sahibi, değeri vb. gibi öznitelik bilgileri arasında bağlantı parsel numarası üzerinden sağlanabilir). Kayıt-tabanlı konumsal veri modelleri, genellikle "Gerçekleştirim veri modelleri" olarak anılan [36] hiyerarşik, ağ, ve ilişkisel veri modellerini kullanır. 1960'lardaki ilk sistemler hiyerarşik veri modelini kullanmakta idi. Hiyerarşik veri modelinin birtakım gereksinimleri karşılayamaması sonucu, ağ ve ilişkisel veri modelleri geliştirilmiştir. İzleyen kısımda bu modellere yer verilmektedir.

1.2.2.1.1 Hiyerarşik Veri Modeli

Hiyerarşik veri modeli, gerçeği hiyerarşik bir yapıda görür. Bu, hiyerarşik karakterdeki gerçeğin gösterimi için oldukça doğrudan ve doğal bir yol olmakla birlikte, hiyerarşik olmayan yapıların gösteriminde problem yaratır. Hiyerarşik yapı genellikle ağaç yapı ile karakterize edilir; Kök, gövde, dal yerleri, dallar, ve yapraklardan oluşan bir yapı. Yabancı literatürde değişik adlarla anılan dal yerleri ve yapraklar, bu tezde sırasıyla "açık-düğüm (iç-düğüm)" ve "kapalı-düğüm (uç-düğüm)" olarak anılacaktır.

Her açık-düğüm (dallanma noktası) ait birden çok kapalı-düğüm bulunmakla birlikte, herhangi bir kapalı-düğüm yalnızca bir açık-düğümüne ait olabilir. Bu nedenle hiyerarşik model ile 1:n ilişkiler doğrudan gösterim edilebilirken, m:n ilişkiler sorun olmaktadır. Örnek olarak poligonlar ve onları oluşturan çizgi parçaları dikkate alınır, poligonun birden çok çizgi parçasından oluşması doğrudan, bir çizgi parçasının birden çok poligona ait olması ise dolaylı olarak gösterim edilebilecektir (Şekil 5a).

Sonuç olarak hiyerarşik veri modeli, hiyerarşik yapıdaki veri için oldukça etkili ve gerçekleştirim açısından da hızlı olmasına karşın, hiyerarşik olmayan ya da hiyerarşik ilişkiler yanında, karmaşık ilişkiler de içeren veri için uygun değildir. O nedenle, hiyerarşik model fiziksel düzeyde daha geniş bir uygulama alanı bulmuştur. Hiyerarşik veri yapıları bugün, verinin fiziksel organizasyonunda yaygın olarak kullanılmaktadır [45].

1.2.2.1.2 Ağ Veri Modeli

Ağ veri modeli ilk olarak 1971 de önerilmiş, daha sonra 1978 ve 1981 de birtakım değişiklikler yapılmıştır [36]. Ağ modelinin temel kavramları "kayıt tipi", "kayıt", "veri elemanı", ve "set" dir. Verinin, bir isim verilebilen en küçük birimi bir veri elemanıdır. Kayıtlar, birbiriyle ilişkili veri elemanlarından oluşur. Kayıt tipi ise, belirli özellikteki kayıtlar için bir kalıp tanımlar. Bir kayıt tipi, kayıt ismi, kayıtları oluşturan veri elemanlarının ismi ve formatı ile tanımlıdır. Bu bakımdan kayıt tipi, tipik bir veri tipi tanımlamasıdır. Parsel örneğine dönülürse, PARSEL varlık tipi burada bir kayıt tipine, parselin özellikleri de veri elemanlarına benzetilebilir. Matematiksel set tanımından farklı olarak, birbirine benzer olmayabilen kayıtlar, bir set oluşturmak üzere birbirine bağlanabilir. Bu bakımdan ağ modeli, hiyerarşik modele göre daha yüksek bir esneklik sunmaktadır.

Ağ modelinin hiyerarşik modelden en önemli farkı, kapalı-düğüm pozisyonundaki verinin açık-düğümüne işaret edebilmesidir (Şekil 5c). Böylelikle ağ modelinde 1:n ilişkiler yanında, m:n ilişkiler de modellenenebilir. Bu da veri tekrarını önemli ölçüde azaltır. Böylece Şekil 5 teki gibi bir durumda, iki poligon arasındaki ortak çizgi ve noktaların hiyerarşik yapının tersine, veri tabanında tek bir yerde tutulması yeterli olacaktır. Bununla birlikte, m:n ilişkilerin gerçekleştirimi genellikle dolaylı yoldan, göstergeçler vasıtasıyla olur [37], [46]. Bu durumda veri tekrarı önlenmekte, ancak veri tabanı bu kez de göstergeçlerle şişmektedir. Ayrıca, göstergeç yapısının ilk başta oluşturulması ve özellikle de devam ettirilmesi oldukça külfetli bir iştir. Çünkü herhangi bir değişiklik durumunda, bütün ilgili göstergeçler yenilenmelidir.

1.2.2.1.3 İlişkisel Veri Modeli

Hiyerarşik ve ağ modellerinin yukarıda söz edilen sorunları, yeni bir model arayışını getirmiş ve [47], İlişkisel veri modelini önermiştir. İlişkisel veri modelinin temel kavramı "ilişki" dir. İlişkiler yardımıyla, veri içerisindeki ilişkiler modellenir. Dolayısıyla ilişkisel bir veri tabanı, çeşitli ilişki örneklerinden oluşur. Örneğin Şekil 5d deki ilişkiler, HARİTA-POLİGON, POLİGON-ÇİZGİ, ve ÇİZGİ-NOKTA dır. Kavramsal olarak ilişkiler, satır ve sütunlardan oluşan iki boyutlu "tablo (ilişki)" larla karakterize edilir. Genellikle veri tabanında her tablo için bir dosya bulunur. Tablonun her satırı birbiriyle ilişkili verilerin bir topluluğudur. Sütunlarda ise nitelikler bulunur. Parsellerden oluşan bir veri tabanında parsellerin PARSEL tablosunda tutulduğu düşünülürse, bu tabloda her parsel için bir satır, parselin alanı, değeri, sahibi vs. niteliklerinin her biri için de bir sütun bulunacaktır. Ağ modeli ile karşılaştırıldığında, tablo satırları kayıtlara, nitelikler ise veri elemanlarına karşılık gelmektedir.

Bir tablo içerisinde satırların sırası önemli değildir ancak, iki satır birbirinin aynı olamaz. Bir tablodaki sütun, yani nitelik sayısı sabittir. Bir tablonun her satırı, bir ya da birkaç sütun değeri vasıtasıyla tek anlamlı olarak tanımlanabilmelidir. Söz konusu sütun değer(ler)i "anahtar" olarak adlandırılır. Anahtarların diğer önemli işlevi, farklı tablolar arasında bağlantıyı sağlamaktır. Bu yolla, kullanıcının veri tabanına yönelttiği bir sorgulamanın cevaplanması için gerekli veri farklı tablolardan alınabilir.

İlişkisel veri tabanı tasarımında tablolar, belli kurallara göre, belli formlara uyacak şekilde oluşturulur. Bu kurallar "normalizasyon kuralları", formlar da "normal form" olarak bilinir [48]. Beş ayrı normal form bulunur ve bunlar birinci, ikinci, vs. normal formlar olarak anılır. İlk üç normal form, [49] in önerdiği formlar olup daha sonra geliştirilen dördüncü ve beşinci formlara göre, çok daha yaygın olarak kullanılmaktadır [37]. Örneğin birinci normal form basit olarak, her bir sütunda tek bir değer bulunmasını öngörür. Buna göre örneğin bir noktanın x ve y koordinatları iki ayrı sütunda tutulmalıdır. Normalizasyon "iyi" bir tasarım için gereklidir ancak kendi başına iyi bir tasarımı garanti etmez [36].

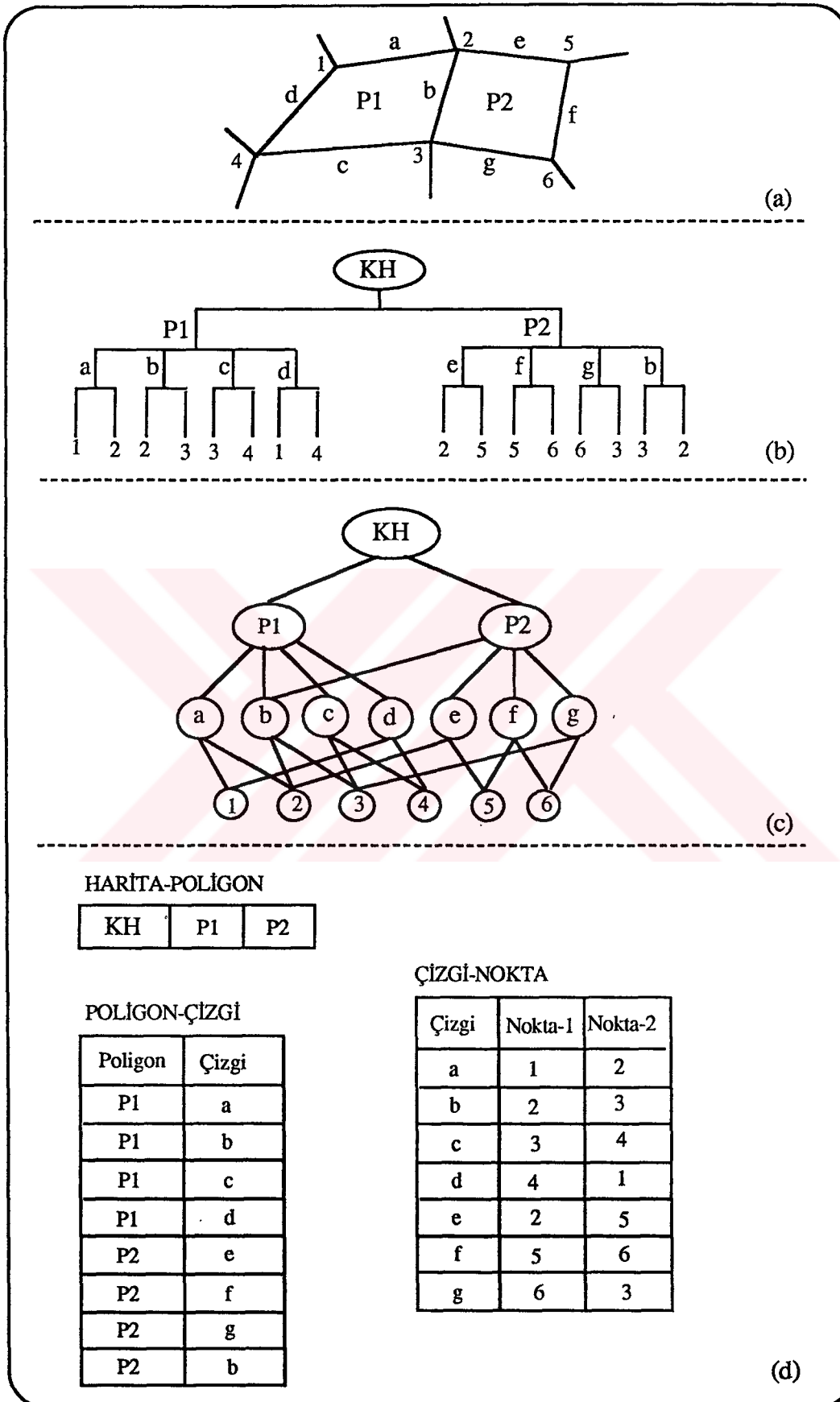
Diğer gerçekleştirim modellerinden farklı olarak, ilişkisel modelde veri içerisindeki ilişkilerin tümü veri tabanı oluşturulurken tanımlanmak zorunda değildir. Bir takım temel ilişkiler oluşturulduktan sonra, istendiğinde bunlardan yeni ilişkiler türetilir. İlişkisel modelin en önemli üstünlüklerinden biri budur. Çünkü farklı kullanıcıların zamana göre değişebilen istek ve gereksinimleri, veri içerisindeki farklı ilişkilere yönelik olmaktadır. Bu bakımdan ilişkisel model, "anlık" (ad-hoc) sorgularla başa çıkmada oldukça güçlüdür.

İlişkisel model, "ilişkisel cebir" ve "ilişkisel hesap", gibi sağlam matematiksel temeller üzerine kurulmuştur. Bu nedenle ilişkisel model çoğu kez "zarif" olarak nitelenir [50]. Gerçekten de ilişkisel modelin popülaritesi çıkışından itibaren hızla artmış, diğer gerçekleştirim modellerine üstünlüğü kabul edilmiş ve konumsal veri yönetiminde bugün de yaygın olarak kullanılmaktadır. Ancak, modelin orjinal olarak karmaşık olmayan veri için geliştirilmiş olması, modeli konumsal veri türünden karmaşık veri içeren uygulamalar (örn. mühendislik uygulamaları) için yetersiz kılmaktadır. Bu nedenle, ilişkisel model için çok çeşitli "genişletme" ler önerilmiş, ve bu genişletmeleri yapısında bulunduran pek çok İlişkisel VTYS (İVTYS) geliştirilmiştir [51], [52], [53], [54], [55].

1.2.2.2 Nesne-Tabanlı Konumsal Veri Modelleri

Geleneksel veri modellerinin anlatım güçlerinin artırılması gereksinimi, semantik veri modelleme alanındaki çalışmaları doğurmuş ve geleneksel veri modelleri için çeşitli anlamsal genişletmeler yanında "semantik veri modelleri" önerilmiştir [36], [51], [52], [53], [55], [56], [57]. Sözkonusu genişletme ve semantik veri modelleri, veri modellemede yeni birtakım kavramlar getirmişlerdir. Çok sayıda semantik veri modeli geliştirilmiş olmakla birlikte, bunların oldukça karmaşık olmaları ve doğrudan gerçekleştirilebilir olmamaları nedeniyle ancak belli başlı bir kaç pratikte uygulama alanı bulmuştur. Başlıca semantik veri modellerine örnek olarak, "Genişletilmiş Varlık-İlişki modeli" [36], "Semantik veri modeli" [58], "Fonksiyonel veri modeli" [59] gösterilebilir. Son yıllarda büyük ilgi toplayan "Nesneye-Dayalı veri modeli" [60] de, semantik veri modelleme alanındaki çalışmalardan esinlenmiştir. Coğrafi Bilgi Sistemleri (CBS) alanında da geniş ilgi uyandıran "Nesneye-Yönelim" tekniğinin CBS'lerde kullanılması, bugün başlıca araştırma alanlarından biri durumundadır.

Semantik veri modelleme alanındaki çalışmaların etkisiyle, son zamanlarda geliştirilen konumsal veri modellerinde, "varlık" kavramı temel modelleme elemanı olarak öne çıkmıştır. Nesne-tabanlı modeller ile veri tabanı yapısı, kayıt-tabanlı modellerden daha yüksek bir düzeyde tanımlanabilir. Veri tabanının konusunu oluşturan gerçek, kendilerine has öznitelikleri ve birbirleriyle ilişkileri olan çeşitli "varlık"ların bir bütünü olarak görülür. Diğer bir anlatımla her varlık öznitelikleri ve diğer varlıklarla olan ilişkileri ile tanımlıdır. Öznitelikler bakımından, konumsal ve konumsal olmayan iki bileşen sözkonusudur. Konumsal bileşen koordinatlarla ya da topolojik olarak ifade edilebilir. Dolayısıyla katman tabanlı modellerin tersine olarak, geometri boyutu varlık içerisinde, onu tanımlayıcı bir unsurdur. Konumsal olmayan bileşen ise, konuma bağlı olmayan özniteliklerdir. Örneğin bir "karayolu" varlığının "konumu", "uzunluğu" gibi konumsal, "şerit sayısı", "son durumu" gibi konuma bağlı olmayan öznitelikleri vardır. Ayrıca bu karayolunun diğer bir



Şekil 5. Bir kadastr haritası ve değişik gerçekleştirim modellerindeki görüntüsü:(a) Kadastr haritası (KH) (b) hiyerarşik görüntü (c) Ağ görüntüsü (d) İlişkisel görüntü.

takım karayolları ile bağlantısı olabilir. Bu durum da bir nesnenin diğer birtakım nesnelerle olan ilişkilerine işaret eder. Nesne-tabanlı modellerde temel modelleme elemanı "varlık" olduğu için, işlemler yüksek bir düzeyde—varlık bazında—ifade edilebilir.

"Nesne-tabanlı" nitelemesi için, yabancı kaynaklarda "detay-tabanlı", "varlık-tabanlı" gibi isimler de kullanılmaktadır [16], [43]. Nesne-tabanlı konumsal veri modelleri için uygun genel amaçlı modeller Şekil 4 de gösterilmiştir. Bu modellerden Varlık-İlişki (Vİ) modeli, en yaygın olarak kullanılan modeldir. Nesneye Dayalı (ND) veri modeli de son yıllarda konumsal veri yönetiminde çok büyük ilgi uyandırmış ve bu modele dayalı pek çok konumsal veri modeli geliştirilmiştir. [35] ve [61] modelleri bunlardan yalnızca ikisidir. Fonksiyonel model pek kullanılmamıştır. Ayrıca semantik veri modeli kavramlarını içeren çeşitli modeller geliştirilmiş fakat bunlar yüksek düzeyli olduklarından, genellikle doğrudan gerçekleştirilememiştir. ND modelin semantik veri modeli kavramlarını da içermesi nedeniyle, izleyen kısımda yalnızca ND veri modeli ve Vİ modeli özetle tanıtılacaktır.

1.2.2.2.1 Varlık-İlişki modeli

Varlık-İlişki (Vİ) modeli ilk defa [62] tarafından önerilmiştir. Vİ modelinin temel kavramları "varlık", "öznitelik", ve "ilişki" dir. Varlık, parsel, yol, kişi gibi fiziksel ya da şirket, proje gibi kavramsal bir varoluş ifade eder. Varlıkları tanımlayan özellikler ise özniteliklerdir. Bir parsel varlığı, büyüklüğü ve değeri gibi öznitelikleri ile tanımlanabilir. Aynı özelliklere sahip varlıklar, bir "varlık tipi" tanımlarlar. Veri tabanındaki bir parsel varlığı, bir parsel tipi örneğidir. Varlık tipi ile varlık örneği arasındaki ilişki, veri tabanı şeması ile veri tabanı örneği arasındaki ilişki gibidir.

Varlıkların birbirleriyle olan ilişkilerinin ifade edilebilmesi, gerçeğin algılandığına en yakın bir biçimde tanımlanabilmesi açısından son derece önemlidir. Çünkü gerçeği oluşturan varlıklar arasında birtakım ilişkiler vardır. Örneğin, iki parselin birbirine ya da yola komşu olması, parselin bir sahibinin olması, bir şirketteki işçilerin belirli projelerde çalışması söz konusu varlıklar arasındaki ilişkilerdir. Varlıklar varlık tipleri ile sınıflanabilirken, ilişkiler için de ilişki tipleri söz konusudur. Bir "ilişki tipi", ilişkinin ismi ve özellikleriyle tanımlanır. İki varlık tipi arasında bir ilişki, yalnızca iki varlık arasında (1:1), tek bir varlıkla birden çok varlık arasında (1:n), ya da karşılıklı ikiden çok varlık arasında (m:n) olabilir. Parsel tipi varlıklarla kişi tipi varlıklar arasında parsel sahiplerini göstermek üzere, bir "parsel-sahibi" ilişkisi düşünülün. Pratikte hiç te olası olmayan bir durum olarak, veri tabanının yalnızca hissesiz parsellerden oluşması durumunda bu ilişki 1:1 (bire bir), aksine yani hisseli parseller söz konusu ise m:n (çoka-çok) bir ilişki tanımlayacaktır. Hisseli

parseller fakat herkesin yalnızca bir parselde hissesi bulunması koşulu ise bu ilişkiyi parselden kişiye doğru 1:n (bire-çok) bir ilişki yapacaktır.

[62], Vİ modeli için grafik gösterim işaretleri de önermiştir. Vİ modeli bugüne kadar genellikle veri tabanı tasarımında, kavramsal tasarım aşamasında kullanılmış, VTYS'lerde gerçekleştirim veri modeli olarak hemen hiç kullanılmamıştır. Bugün Vİ modelinin değişik uyarlamaları mevcuttur. Bunlardan en çok ilgi toplayanı, original Vİ modeli üzerine yeni kavramlar ekleyen "Genişletilmiş Vİ" modelidir [36].

1.2.2.2.2 Nesneye-Dayalı Veri Modeli

Semantik veri modelleme, karmaşık veriyle başa çıkmanın etkili bir yolu olarak kabul edilen "veri soyutlama" tekniğine [63] dayanmaktadır. Veri soyutlama ile yapılan, gerçeği ya da "veri" yi oluşturan veri tiplerinin, gerçeğin görüldüğü çerçeveden tanımlanmasıdır. Veri soyutlama sonucunda elde edilen veri tipleri "Soyut Veri Tipleri (SVT)" olarak anılır. Veri soyutlama ile amaçlanan, gerçeği oluşturan verinin karakteristikleri itibarı ile, alışlagelmiş programlamadaki veri tipleri ile eşdeğer işlevleri olan, çeşitli SVT'lerle karakterize edilmesidir. Yani karakteristikleri bakımından belli bir sınıfa düşen herhangi bir veri, bir SVT ile tanımlanabilir. Böylelikle sözgelimi bir arazi parseli, parsel veri tipinde tanımlanabilir.

Alışlagelmiş programlamada veri tipi, programa giren ya da programın koşumu sırasında oluşan veri değerlerini karakterize etmek için gereklidir. Veri tipi, verinin yapısını ve veriye uygulanabilecek işlemleri belirler. Diğer bir ifadeyle veri tipi, belli özelliklerdeki veri için bir kalıp oluşturur. Örneğin bir veri değerinin tamsayı tipinde tanımlanmış olması, onun tamsayı bir değer olduğunu ve ona çarpma, toplama gibi işlemlerin uygulanabileceğini belirtir. Alışlagelmiş programlama dilleri tamsayı, kesirli sayı, karakter, ve mantıksal gibi oldukça basit ve bazen "ilkel" ya da "temel" veri tipleri olarak anılan sınırlı sayıda veri tipine izin verir.

Genel olarak, veri iki özelliği ile karakterize edilebilir. Bunlar verinin "nitelikleri" ve veriye uygulanabilecek "işlemler" dir. Örneğin parsel tipinde bir veri için konum, alan gibi nitelikler ve alan-hesabı, sınır-düzeltilmesi gibi işlemler sözkonusudur. Nitelikler ve işlemler bazen "hizmetler" olarak anılır [64]. Bir SVT bir grup veri için veri yapısını, alışlagelmiş programlamadaki gibi belli bir gerçekleştirime göre değil de, verinin sunduğu hizmetler ve bu hizmetlerin tabi olduğu özellikler itibarı ile tanımlar [64]. SVT'lerin alışlagelmiş programlamadaki temel veri tipleri üzerine getirdikleri çok önemli iki boyut, "enkapsülasyon" ve "bilgi gizleme" dir [65]. Enkapsülasyon, veri yapısı ve işlemlerin tek

bir birim içinde tutulmasıdır. Bilgi gizleme ise, bir SVT'nin gerçekleştirim detaylarının diğer SVT'lerden gizlenmesidir. Enkapsülasyon, "tekrarlı kod kullanımı"nı teşvik etmektedir. Örneğin yığıt veri yapısı bir SVT olarak tanımlanabilir. Bu tanım yığıtın veri yapısını ve yığıt üzerinde gerçekleştirebilecek yığıta eleman girme, üst elemanı çekme gibi işlemleri kapsayacaktır. Böylece her programcının kendi veri yapısını programında ayrıca tanımlaması yerine tek bir ortak tanımı kullanabilir. Bu çok önemli bir katkıdır ve "yazılım fabrikaları"na giden yoldur [64]. Öte yandan bilgi gizleme, "genişletilebilir yazılım" üretimine olanak tanımaktadır. Çünkü bilgi gizleme ile belli bir SVT'de yapılacak gerçekleştirim değişikliklerinin, diğer SVT'lere etkisi en aza indirilir.

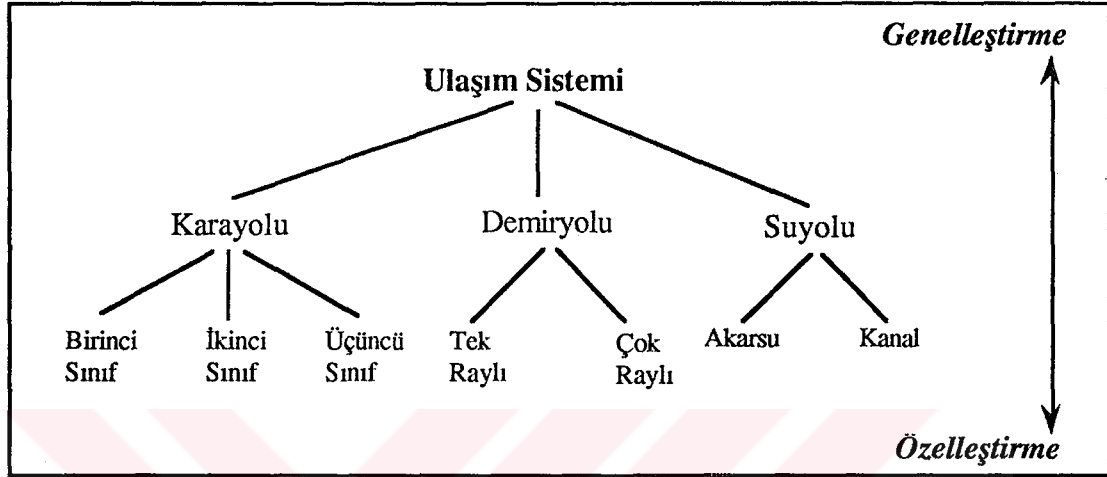
Veri Soyutlama, birtakım soyutlama mekanizmaları yardımıyla gerçekleştirilir. Bunlar "sınıflandırma", "genelleştirme", "birleştirme", ve "ilişkilendirme" dir. Aşağıda bu kavramlar sunulmaktadır.

Sınıflandırma, varlıkların karakteristiklerine göre gruplara ayrılması, yani yukarıda sözedilen gerçeği oluşturan SVT'lerin belirlenmesidir. Dolayısıyla, her grup bir SVT ile tanımlıdır. "Sınıf", ND bir proglama ortamında SVT'lere karşılık gelen gerçekleştirim yapısıdır [64], [65], [66]. Nasıl ki alışlagelmiş bir programlama ortamında çeşitli veri tipi örnekleri yaratılıyor, yani çeşitli tipte değişkenlere veri değerleri atanıyorsa, SVT'lere dayalı bir programlama ortamında da çeşitli "sınıf örnekleri" yaratılır. Bu sınıf örnekleri, ND programlamada "nesne" olarak adlandırılır Böyle bir ortamda programlama, sınıfların sunduğu servisler yardımıyla gerçekleştirilir; Çeşitli tipte sınıf örnekleri, programın amacına göre birbirlerinden çeşitli servisler talep ederler. Benzer şekilde SVT'lere dayalı veri tabanları da sınıf örneklerinden oluşur. Şekil 6 da ulaşım hatlarının bir sınıflaması görülmektedir.

Genelleştirme ile, benzer özellikler taşıyan varlıklar genel bir varlık ile gösterim edilebilirler. Ya da genel varlıklardan daha özel varlıklar tanımlanabilir. Dolayısıyla varlıklar, genelleştirmenin tersi yönde "özelleştirme" bulunan bir hiyerarşi içerisinde algılanır (Şekil 6). Bu hiyerarşi bazen "sınıf hiyerarşisi" olarak anılır [60]. Sınıf hiyerarşisinde en üst ve en alt sınıflar dışında her sınıf için bir "süper" ve bir de "alt sınıf" bulunur. Örneğin Şekil 6 da karayolu, ulaşım_hattı'nın alt sınıfı, birinci_sınıf'ın süper sınıfıdır. Böylelikle genelleştirme, gerçeğin çok genel ve çok özel arasında farklı ayrıntı düzeylerinde görülmesine olanak tanır ki bu da modelleme esnekliği açısından son derece önemlidir.

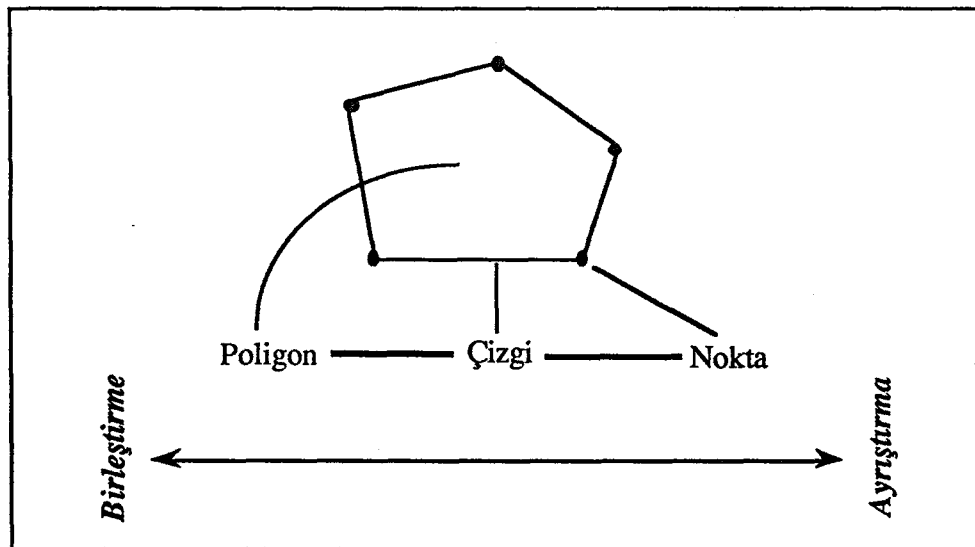
Birleştirme, farklı tipte varlıkların yeni tipte ve daha yüksek düzeyde bir varlık oluşturmak üzere bir araya getirilmesidir. Birleştirme, bir varlık ile onu oluşturan varlıklar arasındaki "oluşum" ilişkisini açıklar. Oluşum ilişkisi içerisinde çeşitli varlıkları kapsamında

bulunduran bir varlık "karmaşık varlık" olarak adlandırılır. Oluşum ilişkisi ile birbirine bağlı sınıfların oluşturduğu hiyerarşi de "sınıf oluşum hiyerarşisi" olarak anılır [60]. Şekil 7 ye göre Poligonlar çizgilerden, çizgiler de noktalardan oluşmaktadır. Buna göre nokta basit, poligon ise karmaşık bir varlıktır. Birleştirme kapsamı, karmaşık varlığı oluşturan varlıkların varoluşlarının, karmaşık varlığın varoluşuna bağlı olduğu bir oluşum ilişkisi ile sınırlı değildir [35], [36]. Diğer bir anlatımla örneğin, çeşitli karayolu varlıklarının, karayolu_ağı



Şekil 6. Genelleştirme ve özelleştirme hiyerarşisi.

gibi yeni tipte, karmaşık bir varlık oluşturmak üzere bir araya getirilmesi de birleştirme kapsamında düşünülür. Söz konusu farkı bir örnekle açıklamak gerekirse, bir haritada poligon sınırlarını oluşturan noktaların yalnızca iki poligona ait olduğu varsayılırsa, bir noktanın ait olduğu her iki poligonun da silinmesi sonucunda, bu noktanın da bir anlamı kalmayacağı için silinmesi gerekir. Ancak bir karayolu_ağı varlığının silinmesi, onu oluşturan karayollarının da silinmesini gerektirmez. Çünkü her bir karayolunun kendi başına da bir anlamı vardır.



Şekil 7. Birleştirme ve ayrıştırma kavramları.

İlişkilendirme ile, varlıklar arasında gerçekte varolan bir takım ilişkiler gösterim edilir. Örneğin, iki karayolunun birbirine bağlı olması ya da bir parselin bir caddeye cephesinin olması ilişkilendirme yapısı ile gösterim edilebilir. ND veri modellemesinde bu, tıpkı bir sınıf gibi tanımlanabilen ilişki tipleri ile olur [35]. Birleştirmeden farkı, birbirlerine ilişkilendirilmiş varlıkların birbirinin parçası olmak zorunda olmamasıdır [35]. Diğer bir anlatımla, birleştirmede karmaşık varlığı oluşturan bir varlığın varoluşu karmaşık varlığa bağlı olabilirken, ilişkilendirmede böyle bir durum sözkonusu değildir [36]. Örneğin iki karayolunun birbirine bağlı olduğu ilişkisinin iptal edilmesi, ilgili karayollarının silinmesini gerektirmez.

Genel olarak bu dört soyutlama mekanizması semantik veri modelleri kapsamında da bulunmaktadır [36]. Ancak ND veri modelinin bunların üzerine getirdiği çok önemli bir boyut, soyutlamanın yalnızca nitelikler bazında olmayıp, işlemlerin de dikkate alınarak yapılmasıdır. Bu bakımdan baştaki SVT tanımlaması daha çok ND modele uygun bir tanımlamadır.

1.2.3 Fiziksel Veri Modelleri

Fiziksel veri tabanı tasarımında, verinin veri tabanında fiziksel olarak depolanması için kullanılacak dosya yapıları ile veri tabanına erişim yol ve yöntemleri belirlenir. Fiziksel tasarım sonunda elde edilen iç şema, seçilen fiziksel modele göre dosya, kayıt, ve kayıt alanlarının nasıl organize edildiğini ve birbiriyle nasıl ilişkili olduğunu ve veriye erişim yollarını gösterir. Genellikle bir VTYS, dosya organizasyonu ve veri erişimi için bir dizi seçenek sunar. Bunlar, çeşitli indeksleme tipleri, birbiriyle ilişkili kayıtların göstergeçler yardımıyla birbirine bağlanması, çeşitli "adresleme" teknikleri, birbiriyle ilişkili kayıtların disk üzerinde "gruplanması" dır. Uygun veri yapılarının seçiminde amaç, iyi performans ve bilgisayar kaynaklarının en iyi şekilde kullanımıdır [36], [37]. Bunun için genel olarak yanıt zamanı, depolama alanı gereksinimi, ve işlem yoğunluğu gibi ölçütlere göre bir değerlendirme yapılır. "Yanıt zamanı", bir işlemin aldığı zamandır. Depolama alanı gereksinimi, belli bir dosya yapısına göre dosyalar ve erişim yolları için gerekli disk alanıdır. İşlem yoğunluğu ise, dakikada gerçekleştirilebilen ortalama işlem sayısıdır [36].

Fiziksel veri modelleri çoğunlukla "veri yapıları" olarak anılır [37]. "Dosya yapısı" dendiğinde ise, bir veri yapısının gerçekleştirimi anlaşılır [39]. Veri yapısı ve dosya yapısı terimleri arasındaki ayırım önemlidir, çünkü aynı bir veri yapısı değişik dosya yapıları kullanılarak gerçekleştirilebilir. Örneğin bir liste veri yapısı, sıralı erişimli olarak ya da göstergeçler kullanılarak gerçekleştirilebilir. Tıpkı kavramsal veri modellerinde olduğu gibi, "diziler", "liste", "yığıt", "kuyruk", "ağaç" gibi genel amaçlı veri yapılarının konumsal veri

için yetersiz kalmaları nedeniyle, konumsal veri için çeşitli veri yapıları geliştirilmiştir. Konumsal veri yapıları genellikle hiyerarşik, yani ağaç yapılardır. En çok bilinen konumsal veri yapıları olarak "k-d ağaç", "dörtlü-ağaç", "sekizli-ağaç" [45] sayılabilir.

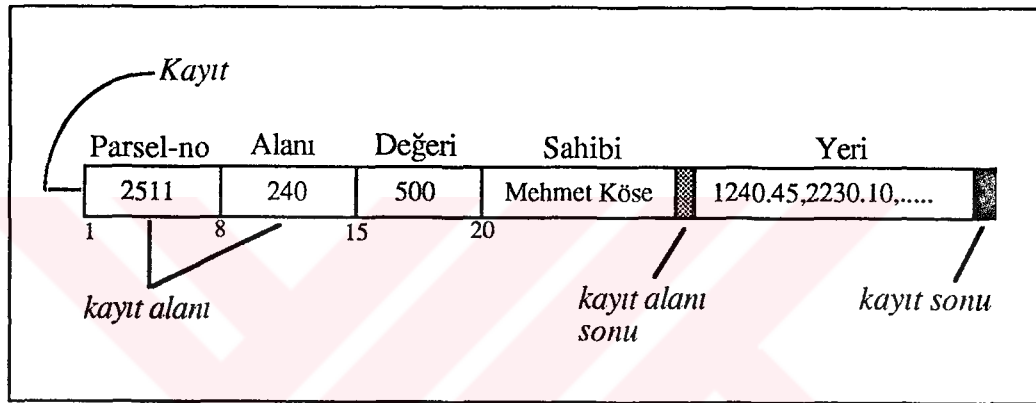
Yukarıda belirtildiği gibi, transfer veri grubunda yer alan verinin formatı, kullanılan fiziksel veri modeli ile ilişkilidir. Örneğin bir coğrafi bölgeye ait noktasal verinin sayısal gösterimi ele alınsın: Bunun için çeşitli veri yapıları kullanılabilir, örneğin coğrafi bölge düzenli bölümler şeklinde düşünülüp her bölümlere içerisindeki noktalar kaydedilebilir. Ya da nokta yoğunluğunun coğrafi bölge içerisinde farklılık göstermesi durumunda verim açısından daha uygun olan, hiyerarşik bir bölümlere uygulanabilir. Bu durumda bölümlere, düzenli bölümlenimin tersine olarak belli bir koşula sağlanıncaya kadar (örn. bir bölüm içindeki nokta sayısına) sürer (Bu konuya 5.1.2 de daha geniş olarak yer verilmektedir). Bu ikinci durumda da "dörtlü-ağaç" veri yapısı kullanılabilir. Dahası, dörtlü ağaç veri yapısı değişik dosya yapıları kullanılarak gerçekleştirilebilir. Örneğin hiyerarşik yapı göstergeçler kullanılarak gösterim edilebilir ya da bölümlere atanacak kod değerleri bir "sıralı liste" de tutulabilir. Bütün bunlar transfer veri grubunun formatını etkileyebilir. Bu durumun daha iyi anlaşılabilmesi için izleyen kısımda, önce konunun bütünlüğü açısından çok genel olarak dosya yapıları ve dosya organizasyonuna değinilecek, ardından da konumsal veri yapısı ile sayısal veri gösterimi arasındaki ilişki gösterilecektir. Fiziksel veri modelleri ve konumsal veri yapıları hakkında daha ayrıntılı bilgi için, bu bölümde anılan kaynaklara başvurulabilir.

1.2.3.1 Dosya Yapıları ve Erişim Teknikleri

Genel olarak bir dosya kayıtlardan, kayıtlar ise veri elemanlarından oluşur. "Veri elemanı", verinin tanımlanan en küçük birimidir. Bir parselin alan değeri gibi. Her veri elemanı, bir veya birkaç Byte büyüklüğünde olabilen bir "kayıt alanı"nda tutulur. Bir "Byte" bilgisayarda en küçük depolama birimi olan "bit"lerden oluşur. Byte büyüklüğü donanım bağımlı bir kavram olmakla birlikte tipik olarak 4-8 bittir. Byte yerine yaygın olarak kullanılan diğer bir sözcük "karakter" dir [36]. Her kayıt alanındaki veri, bir "veri tipi" ile tanımlıdır. "Kayıt tipi" ya da "kayıt formatı" ise, kayıtları oluşturan alanların isimleri ve veri tipleri ile tanımlanır (Şekil 8). Ağ veri modeli açıklanırken bu kavramlardan bazılarını kısaca değinilmişti.

Kayıt tipi, aynı özellikteki kayıtlar için bir kalıp tanımlar, diğer bir anlatımla, aynı kayıt tipindeki kayıtlar aynı alanlardan oluşur. Kayıtlar sabit ya da değişken uzunlukta olabilirler. "Sabit-uzunluklu" kayıtlar sabit uzunluktaki kayıt alanlarından oluşur. Sıkça karşılaşılan bir durum olarak, belirli bir kayıt alanında yer alacak veri uzunluğunun değişebilmesi, kayıt

alanı uzunluğunun sabit tutulmasını engeller. Örneğin parsel sahiplerinin isimlerine ait kayıt alanının sabit bir uzunlukta tanımlanması, uzun isimlerin kesilmesini gerektirirken, kısa isimler için depolama yeri israfına yol açacaktır. Aynı şekilde, parselin yerinin parsel sınır koordinatlarıyla tanımlanması durumunda, bu alanın büyüklüğünü sabitleştirmek çok zor olacaktır (Şekil 8). Bu durumda değişken uzunluktaki kayıt alanlarından oluşan "değişken-uzunluklu" kayıtlar kullanılır. Değişken-uzunluklu kayıt gerektiren diğer bir durum kayıtların "seçmeli" alanlara sahip olmasıdır. Bir kayıt alanının seçmeli olması, bazı kayıtlarda bu alan için bir veri değeri bulunamayabileceği anlamına gelir. Örneğin koordinat değerlerinden oluşan bir kayıta "z" koordinatı seçmeli olarak belirtilmişse, bazı kayıtlarda yalnızca "x" ve "y" değerleri bulunabilecektir.



Şekil 8. Bir parsel için kayıt: İlk üç kayıt alanı sabit-uzunluklu, son iki alan ise değişken-uzunlukludur. Kayıt alanı ve kayıt sonları için, gösterim kolaylığı açısından şekildeki gösterim biçimi tercih edilmiştir.

Aynı tipte sabit uzunluklu kayıtlardan oluşan dosyalar bazen "sabit-uzunluklu dosya" olarak anılır. "Değişken-uzunluklu dosya" dendiğinde, aynı tipte değişken uzunluklu kayıtlardan, ya da farklı tipte kayıtlardan oluşan dosya anlaşılır. Sabit-uzunluklu bir dosyada kayıt alanı uzunlukları sabit olduğundan, herhangi bir kayıt alanının kayıt başlangıcına göre, başlangıç Byte pozisyonu bellidir. Değişken-uzunluklu dosyalarda ise kayıt alanı ve kayıt sınırları bir şekilde tanımlanmalıdır. Bunun için genellikle, dosyada bulunması olası olmayan özel karakterler (\$, %, & gibi) kullanılır.

Bir dosyayı kullanan programlar onun yapısını bilmek zorundadır. Dosya yapısı ile ilgili bilgiler, dosya tanımlayıcısında bulunur. "Dosya tanımlayıcısı", dosyanın disk adresi ve kayıt formatı bilgilerini kapsar. Sabit-uzunluklu dosyalar için, kayıt alanı uzunlukları ve kayıt alanlarının sırası kayıt formatını tanımlamaya yeterlidir. Değişken-uzunluklu dosyalarda ise, kayıt alanı veri tipi kodları, kayıt tipi kodları ile kayıt alanı ve kayıt sonu karakterleri kayıt formatını tanımlar.

Yukarıda da belirtildiği gibi dosya organizasyonunda amaç iyi performans ve bilgisayar kaynaklarının en iyi bir şekilde kullanımıdır. Bunun için veri dosyaları kendi başlarına uygun şekilde düzenlenebileceği gibi, veri dosyalarına ek olarak, tamamen kayıt erişim hızını artırmaya yönelik bir takım yardımcı dosyalar da (indeks dosyaları gibi) oluşturulabilir. Bu bakımdan veri dosyaları "düzenli" ve "düzensiz" olmak üzere iki grupta düşünülebilir. Düzensiz dosyalarda kayıtlar ilk kaydedildikleri gibi rastgele bir sırada dosyada yer alırlar. Düzenli dosyalarda ise, kayıtlar belli bir kayıt alanındaki değerlere göre düzenlenmiş (sıralanmış) tır. Parsel kayıtlarının parsel numaralarına göre sıralanması gibi.

Yabancı literatürde dosya yapılarının sınıflandırılmasında yaklaşım ve terminoloji farklılıkları görülmektedir. Kavram kargaşasına yol açan bu durumu önlemek açısından, dosya yapısı ve erişim tekniği yerine, dosya organizasyonuna göre bir sınıflamanın daha yararlı olacağı düşüncesiyle bu tezde aşağıdaki sınıflama önerilmiştir.

Sıralı Erişimli Dosya Organizasyonu : Sıralı erişimli dosya organizasyonunda kayıtlar birbiri ardına, herhangi bir şekilde sıralanmaksızın dosyaya kaydedilir. Bu durum istenen bir kayıta erişebilmek için, dosya başından başlayarak, aranan kayıta ulaşıncaya kadar sıra ile, bütün kayıtlara bakılmasını gerektirir. Bu yüzden bu erişim şekli "sıralı erişim" olarak adlandırılır. "k" sayıda kayıt için bu arama işlemi, ortalama durumda yani aranan kayıtların dosyanın ortasında bulunması durumunda " $k/2$ ", en kötü durumda yani kayıtların dosyada bulunmaması ya da dosyanın en sonunda bulunması durumunda da "k" sayıda erişimi gerektirecektir. Dolayısıyla veri erişim hızı düşüktür. Buna karşılık dosyaya yeni veri girme işlemi çok basit bir şekilde, verinin dosya sonuna eklenmesiyle yapılır. Veri silme işlemi de arama gerektirdiğinden yavaştır.

İndeksli Dosya Organizasyonu : İndeksli dosya organizasyonunda veri dosyasına ek olarak indeks dosyaları bulunur. İndex dosyalarının işlevi ders kitaplarındaki indekslerle aynıdır. İndeksler, veri dosyasında belirli bir kayıt alanı değerlerine göre düzenlenir. Bu alan, "indeksleme alanı" olarak adlandırılır. Buna göre, indeks dosyasındaki kayıtlarda iki alan bulunur; Bu alanların birinde indeksleme alanı değeri, diğerinde ise veri dosyasında bu değeri taşıyan kayıtların fiziksel adresi yer alır. İndeks dosyaları, indeksleme alanı değerlerine göre sıralanmış bir yapıdadır. Bunun amacı, "ikili arama" tekniği ile arama zamanını kısaltmaktır. Böylece herhangi bir sorgulama için indeksleme alanı değeri verilmişken, ilgili kayıta sıralı erişime oranla çok daha kısa bir zamanda ulaşılabilir. Bu erişim şekli çoğunlukla "rastgele erişim" ya da "doğrudan erişim" olarak adlandırılır [67].

İndeksli dosya organizasyonunda veri dosyaları, indeksleme alanı değerlerine göre düzenli ya da düzensiz olabilir. İndeksleme alanı, anahtar bir alan olabileceği gibi, anahtar

olmayan bir alan da olabilir. Anahtarların özellikleri kayıtları tek anlamlı olarak tanımlamalarıdır. Diğer bir anlatımla, verilen bir anahtar değeri için veri dosyasında yalnızca bir kayıt vardır. Örneğin parsel kayıtlarından oluşan bir dosyada, parsel numaraları anahtar olarak alınabilir. İndeksleme alanının anahtar alan olup olmaması ve veri dosyasının düzenli olup olmamasına bağlı olarak, indeksler "sığ" ya da "seyrek" olabilir [36]. Sığ bir indeks dosyasında veri dosyasındaki her kayıt için bir giriş bulunur. Seyrek indeks dosyalarında ise veri dosyasında birden çok kayıttan oluşan blokların her biri için bir giriş bulunur. Bu durumda aranan bir kayıt için önce ilgili blok kazanılır ve kayıt o blokta aranır.

Arama zamanını daha da kısaltmak için indeks dosyalarının kendileri için de ayrıca indeks dosyaları oluşturulabilir. Bu indeksler "çok düzeyli indeksler" olarak anılır. Bu durumda, zaten zor olan indeks oluşturma ve yenileme işlemleri daha da güçleşecektir.

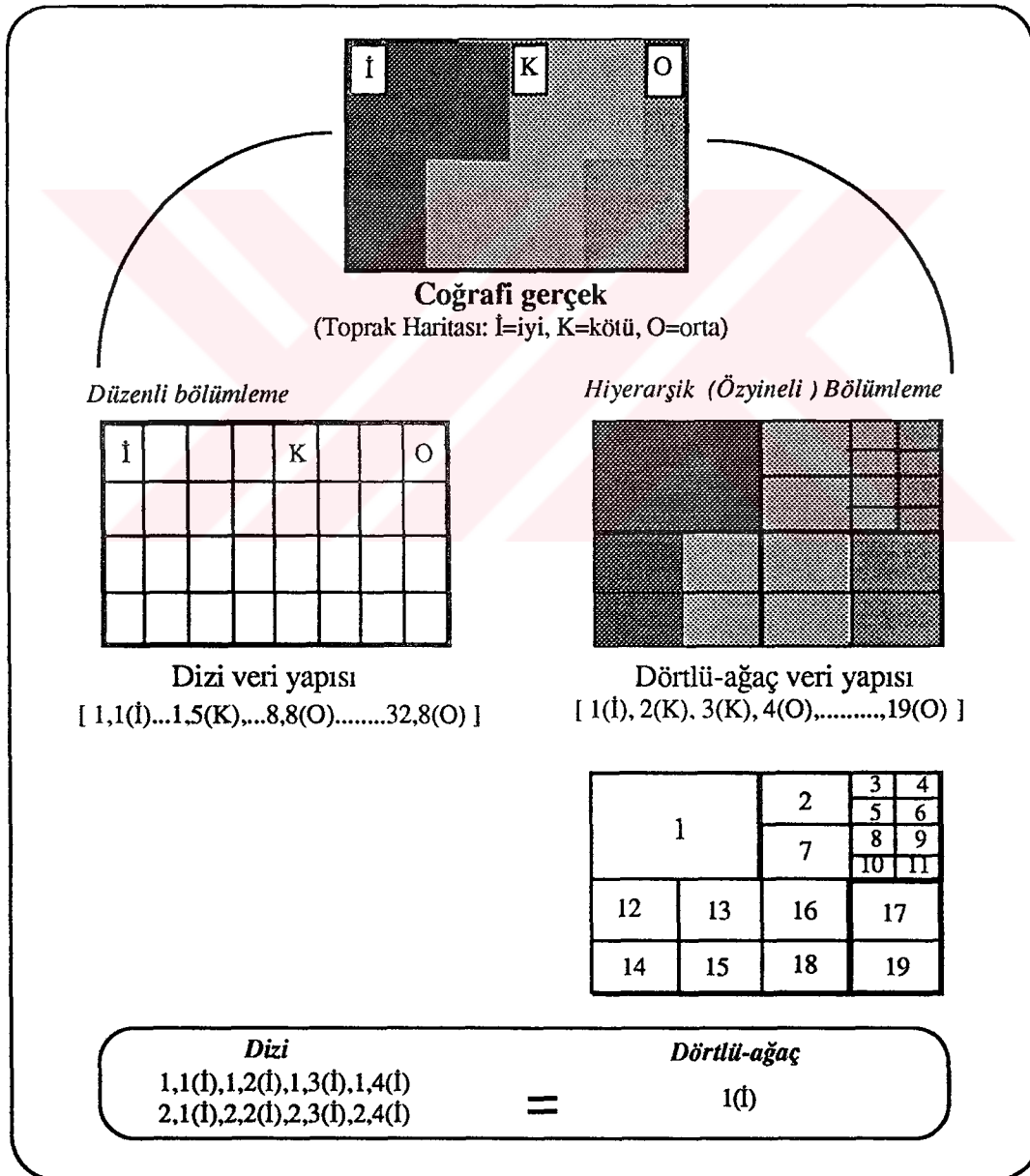
Adreslemeli Dosya Organizasyonu : Doğrudan erişim sağlayan diğer bir dosya organizasyonu, adreslemeli dosya organizasyonudur. "Adresleme", matematiksel bir fonksiyon yardımıyla bir kayıt alanı değerinden, kaydın adresinin belirlenmesidir. Sözkonusu alan "adresleme alanı", matematiksel fonksiyon da "adresleme fonksiyonu" olarak bilinir. Adresleme alanı, çoğunlukla olduğu gibi eğer anahtar bir alan ise, o durumda adresleme anahtarı olarak anılır [36]. Bir adresleme fonksiyonu, genellikle kaydın bulunduğu blok adresini verir. Blok içinde aranan kayıta erişebilmek için ayrıca bir arama gerekir.

Çok çeşitli adresleme fonksiyonları vardır. Adresleme fonksiyonu seçiminde ölçüt, her bloğa mümkün olduğunca eşit sayıda kayıt düşmesinin sağlanmasıdır. Adresleme alanı değerlerinin tamsayı değerler olduğu ve N blok bulunduğu varsayılırsa, adresleme fonksiyonu olarak $h(T) = T \bmod N$ seçilebilir. Bu fonksiyon T tamsayı değerinin N'ye bölümünden kalanı verir ki bu da kayıt adresi olarak kullanılır. Kayıtların bloklara homojen bir şekilde dağılımının sağlanması, çoğu adresleme fonksiyonları için problem yaratır. Genel olarak adresleme alanının alabileceği değerler, mevcut kayıtlardan çok daha fazladır. Sonuçta adresleme fonksiyonu iki farklı kayıdı aynı adrese atayabilecektir. Bunun önlenmesi ve kayıt adreslerini yeni kayıtlar eklemek üzere artırmak için çeşitli teknikler mevcuttur. Tezin ilgi alanı gereği bu ayrıntılara girilmeyecektir. Bu konuda daha fazla bilgi için [36], [68] den yararlanılabilir.

1.2.3.2 Konumsal Veri Yapısı ve Veri Formatı Arasındaki İlişki

Konumsal veri yapısı ve veri formatı arasındaki ilişkinin gösterilmesi için şekil 9 daki toprak haritasının sayısal olarak gösterilmesi ele alınsın. "Raster" formda gösterilmesi düşünülen bu toprak haritası için, çok çeşitli gösterimler söz konusu olabilir. Bunlardan biri,

toprak haritasını sınırlayan coğrafi bölgenin toprak kalitesinin dağılımı dikkate alınmaksızın, sabit boyutlu bölmeler elde edecek şekilde bölümlenmesi ve her bir bölme (göze) karşılık gelen toprak kalitesi değerinin kaydedilmesidir. Diğer gösterim şekli ise, bölümlenmenin toprak kalitesinin dağılımı dikkate alınarak yapılması, bölümlenmenin her bir bölme için tek bir değer elde edilinceye kadar sürmesi ve yine her bir bölme karşılık gelen değerlerin kaydedilmesi ilkesine dayanır. Bu ikinci gösterim için özyineli bir bölümlenme uygulanacağı için hiyerarşik bir yapı ortaya çıkacak değişken boyutlu bölmeler elde edilecektir. Bu gösterim şekli için şekil 9 da, dörtlü ağaç yapıya göre bölümlenme uygulanmış, ve bölmeler "Z" harfi yönünde—kuzey batı "0", kuzey doğu "1", güney batı "2", ve güney doğu "3"—numaralandırılmıştır.



Şekil 9. Bir toprak haritasının farklı veri yapıları kullanılarak gösterimi.

Birinci gösterim şekli, "dizi" veri yapısı kullanılarak gerçekleştirilebilir. Buna göre, dizinin her bir satır ve sütunu için bir değer bulunacaktır. Örneğin dizinin 1. satır 5. sütun elemanının değeri, "K" dır. İkinci gösterim şekli için kullanılan dörtlü ağaç yapı farklı şekillerde gerçekleştirilebilir [45]. Örneğin yukarıda da belirtildiği gibi, göstergeçler kullanılabilir. Bu durumda açık düğümler kapalı düğümlere göstergeçler bulundurur. Kapalı düğümler yalnızca düğüm değerini saklar, göstergeç bulundurmaz. Genel olarak, dörtlü ağaç yapı için diğer gerçekleştirim şekli, göstergeçsiz her bölme için bir kod değeri üretilebilir ve bu kod üzerinden düğüm (bölme) değerine ulaşılır. Bu kod değerlerinin sıralı bir listede tutulduğu gerçekleştirim şekli "doğrusal dörtlü ağaç" olarak anılır.

Tezin ilgi alanı gereği burada daha fazla ayrıntıya girmeye gerek yoktur. Burada altı çizilmek istenen, veri yapılarının veri formatını etkileyeceğidir. O nedenle bu durum şekil 9 da, alt kısmındaki pencerede yalnızca şematik bir biçimde gösterilmiştir. Görüleceği üzere, Dörtlü ağaç gerçekleştirim şekli ne olursa olsun, toprak haritasının dizi veri yapısı ve dörtlü ağaç veri yapısı kullanılması durumundaki sayısal gösterimleri arasındaki fark oldukça belirgindir.

1.3 Konumsal Veri Değişimi

Konumsal veri değişimi, aynı bir gerçeğin farklı sayısal gösterimleri arasında bir çeviri gerektirir. Konumsal veri, herhangi bir sayısal gösteriminde, belirli bir "format"a, yani sözdizime uymakta ve bir anlam taşımaktadır. Dolayısıyla, konumsal veri değişiminde sözkonusu olan, sözdizimsel ve aynı zamanda da anlamsal bir çeviridir. Anlam bir gerçeğin nasıl, "algılandığı", bir başka deyişle hangi çerçeveden "görüldüğü" ile ilişkilidir. Bir konumsal veri değişimindeki tarafların belli bir gerçeği algılayışları birbirine ne kadar yakınsa, çeviri de o kadar kolay olacaktır. Konumsal veri için çok çeşitli veri modellerinin kullanılması, konumsal veri değişimini oldukça karmaşık bir işlem yapmaktadır [11], [12].

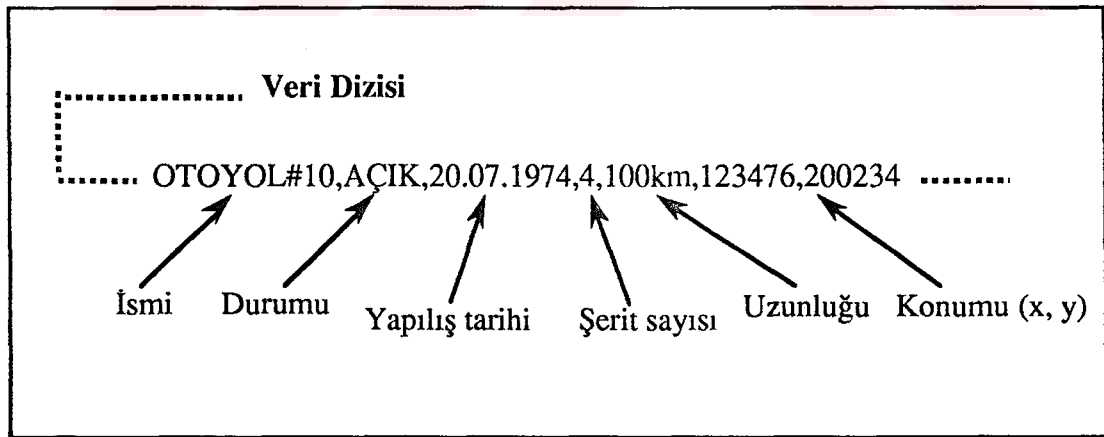
Konumsal veri değişimi, bir formattan diğerine "doğrudan" ya da bir ara format üzerinden "dolaylı" olarak gerçekleştirilebilir. Genel amaçlı olarak geliştirilmiş ara formatlar, "değişim formatı" olarak anılır. Pratikte daha yaygın kullanım alanı bulmuş olan, dolaylı yöntemdir. Bununla birlikte, konumsal veri değişimi, bugüne kadar özel amaçlı çözümlerle ele alınmış olmasından ötürü, bugün arzu edilen bir düzeyde gerçekleştirilememektedir [11], [13], [14], [15], [16]. Bu da aslında, değişim formatı tasarımının hep ön planda tutulmuş olması, gerçekleştirim boyutunun ise adeta gözardı edilmesidir. Gerçi, değişim formatı tasarımı alanında önemli gelişmeler sağlanmıştır. Örneğin, SAIF (Spatial Archive and Interchange Format) [35] ve SDTS (Spatial Data Transfer Standart) [22] gibi yeni değişim formatları, veri modelleme alanındaki yeni gelişmeleri kullanmakta ve ISO'nun

(International Standart Organization) OSI (Open Systems Interconnected) modelini [69] desteklemektedirler. Ancak, konumsal veri değişiminin pratikte uygulanışı ele alındığında, veri gruplarını görme, tarama, sorgulama ve ardından da istenen alt grupları ya da detayların değişimine olanak tanıyacak kullanıcı arabirimlerinin eksikliği göze çarpmaktadır [15].

Konumsal veri değişiminin iki boyutu vardır. Bunlardan biri çeviri, diğeri iletişim boyutudur. Çeviri boyutu sunucu, alıcı, ve ara formatlar arasındaki çevirilere işaret eder. İletişim boyutu, taraflar arasındaki iletişimi ve verinin bir taraftan diğerine transferini içermektedir. Bu kısmın birinci alt kısmı çeviri ve iletişim boyutlarının açıklanmasına ayrılmıştır. İkinci alt kısımda ise doğrudan ve dolaylı konumsal veri değişim yöntemleri ele alınmaktadır.

1.3.1 Konumsal Veri Değişiminin İçeriği

Bir sayısal veri gösteriminde, veri belirli bir sözdizimine uyar ve bir anlam taşır. "Anlam", belli bir gerçeğin nasıl algılandığı ya da hangi çerçeveden görüldüğü ile ilişkilidir. "Sözdizim" ise, bu algılamamanın sayısal olarak gösterim şeklini belirler ve çoğunlukla verinin "formatı" olarak anılır. Örnek olarak, bir sayısal veri dizisinde bir OTOYOL'a ait *şerit sayısı* değerinin, yolun *yapılış tarihi* değerinden sonra gelmesi sözdizimin belirlediği bir özelliktir. Öte yandan, "4" gibi bir değer yolun şerit sayısını gösterdiği ve bu özniteliğin pozitif bir tamsayı değeri alması gerektiği gerçeği ise anlamsal bir özelliktir (Şekil 10).



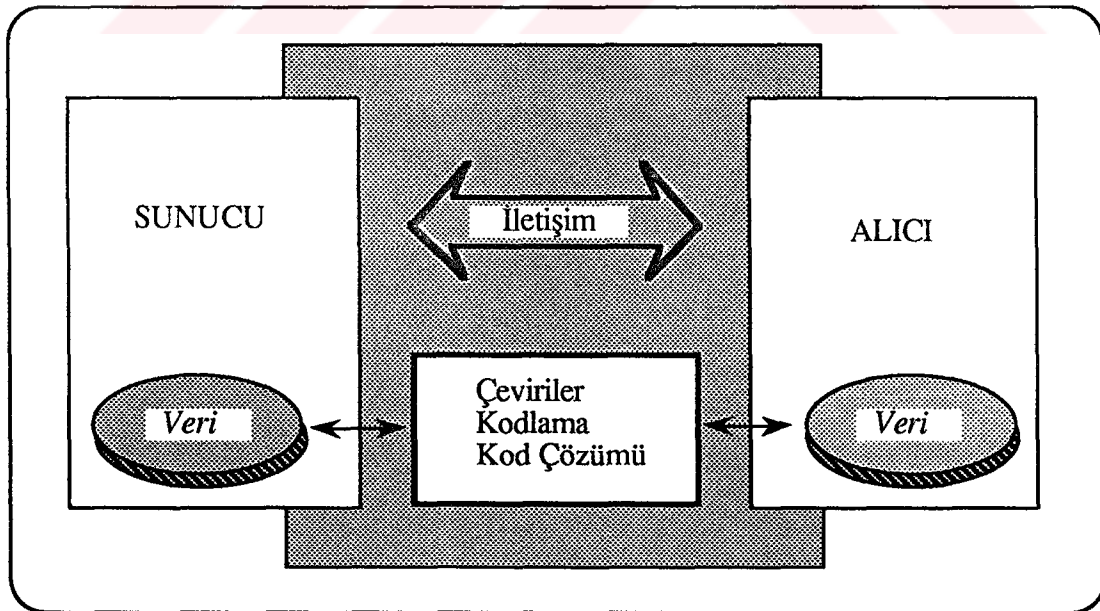
Şekil 10. Bir veri dizisinde sözdizim ve anlam.

Konumsal veri değişimi, verinin iki farklı formattaki gösterimleri arasında bir çeviri gerektirir. Bu çeviri hem sözdizimsel, hem de anlamsaldır. Şöyle ki, bir veri grubundan diğerine bir transfer yapabilmek için, her veri elemanının bir tarafta "ne anlama geldiği", ve diğer tarafta "neye karşılık geldiği"nin belirlenmesi gerekir. Söz konusu çeviri, konumsal

veri deęişimini gerekleřtiren tarafların aynı bir gereęi farklı algılamaları ve gstermelerindendir.

Konumsal veri deęişimini gerekleřtiren taraflardan biri "sunucu", dięeri ise "alıcı" dır. Sunucu veriyi saęlayan, alıcı ise veriyi transfer eden taraftır. Sunucu ve alıcı, iki KVYS olabileceęi gibi, alıcı bir KVYS, sunucu ise veri saęlayıcı bir kuruluş olabilir. Konumsal veri deęişimi, sunucu formatındaki verinin alıcı formatına evirilmesi ve alıcıya iletilmesini ierir. Konumsal veri deęişimi alıcının, sunucudan veri istemesi zerine gerekleřir. Dolayısıyla, konumsal veri deęişiminin iki boyutu vardır; "eviri" ve "iletiřim" (řekil 11). Sunucu formatındaki verinin alıcı formatına dnřtrlmesi eviri boyutu, alıcının "hangi" veriye gereksinim duyduęunu karřı tarafa bildirmesi ve bu verinin bir řekilde alıcıya iletilmesi de iletiřim boyutu kapsamındadır.

eviriler aynı verinin her iki tarafta farklı řekillerde algılanması ve gsterilmesi sonucu oluřan farklılıklar nedeniyledir. Sunucu formatındaki verinin alıcı formatına dnřtrlmesi, bir dizi eviri gerektirebilir [70]. eviri boyutu aynı zamanda verinin alıcıya transferi iin belirli bir formatta kodlanmasını ve kodlanmış verinin alıcı tarafından zmlenmesini de kapsamaktadır. Konumsal veri deęişiminin eviri ve iletiřim gibi iki ayrı boyutta dřnlmesi, ISO OSI modeli [71] ile uyumludur. Buna gre eviri boyutu sunuř ve uygulama dzeyine, iletiřim boyutu uygulama ve 1 den 5 e kadar olan dzeylere iřaret etmektedir. eviri konusu, 1.3.2.1 de ele alınmaktadır.



řekil 11. Konumsal veri deęişiminin ierięi.

Konumsal veri deęişiminde iletiřim, tarafların birbirini anlayabilmesi bakımından son derece önemlidir [16], [21]. Alıcının hangi veriyi istedięini sunucuya bildirmesi, ve buna g re gerekli verinin alıcıya iletilmesi, iletiřim boyutu kapsamındadır. İletiřim, " evrim-dıřı" ya da " evrim-i i" bir tarzda ger ekleřtirilebilir. İletiřim konusu 1.3.2.2 de ele alınmaktadır.

Konumsal veri deęişiminde, " evirici" dendięinde  eviri, kodlama, ve kod   z mlleme iřlemlerini yapan program anlařılır. Dolayısıyla bir  evirici programı  eviri, kodlama ve kod   z mlleme mod llerini i erir.

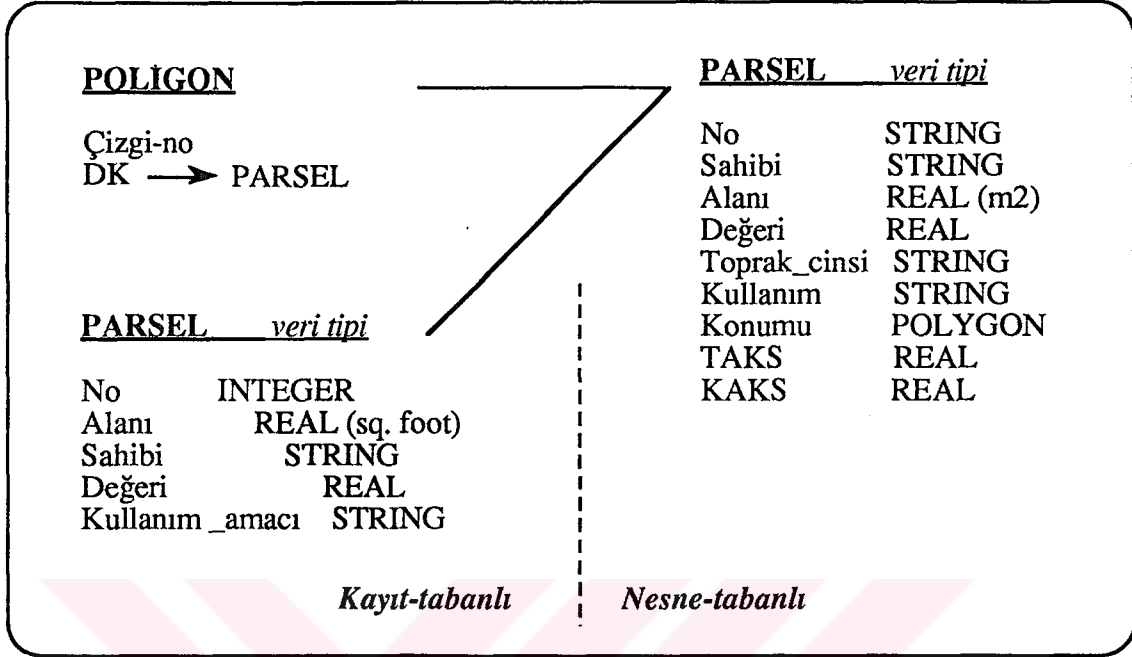
1.3.1.1  eviriler

 eviriler, konumsal veri deęişiminde bulunan tarafların aynı bir ger eęi kavramsal olarak algılamalarında ve sayısal olarak g sterim etmelerindeki farklılıklar nedeniyledir. İkinci b l mde belirtildięi gibi, bir ger eęin belirli bir perspektifte algılanarak tanımlanması i in veri modelleri kullanılır. Veri modellerinin saęladığı kavram ve kurallar yardımıyla ger ek, belirli gereksinim ve beklentilerin belirledięi bir algılama perspektifinde soyutlanarak, insan ve bilgisayarın anlayacaęı d zeylerde tanımlanır. Bu d zeyler kavramsal ve fiziksel d zeyler, s z konusu tanımlamalar kavramsal ve i  řemalardır.

Farklı gereksinim ve beklentiler, aynı bir ger eęin farklı řekillerde algılanmasına neden olur.  rneęin bir grup konumsal veri kullanıcısı, imar planı ve topoęrafik yapı verilerine gereksinim duyarken, dięer bir grup i in topoęrafik yapı ve tesis hatları (su, kanalizasyon, doęal gaz, enerji, telefon)  nemli olabilir; Bir grup i in hi   nem tařımayan m lkiyet bilgileri, dięer bir grup i in  ok  nemli olabilir; Ya da iki farklı kullanıcı grubunun parsel tanımlamaları olduk a farklı olabilir. Benzer řekilde, ger ekleřtirim a ısından farklı beklentiler olabilir; Bir grup i in depolama alanı  nemli iken, dięeri i in iřlem hızı daha  nemli olabilir.

 zetle, konumsal veri deęişiminde bulunan taraflar arasında veri modeli, řema, ve ger ekleřtirim bazında farklılıklar olabilir; Taraflar farklı veri modelleri, ya da aynı veri modeli fakat farklı řemalar kullanabilir, ve b y k olasılıkla farklı ger ekleřtirimler uygulayabilirler. Veri modeli farklılıklarına  rnek olarak, řekil 12 deki gibi bir tarafın kayıt-tabanlı dięer tarafın ise nesne-tabanlı bir model kullanması verilebilir. řema farklılıklarına  rnek olarak řekil 12 deki parsel tanımlamaları arasındaki farklılıklar ( rn. kayıt-tabanlı tarafta toprak cinsine ait bir bilginin olmaması) g sterilebilir. Ger ekleřtirim farklılıklarına  rnek olarak bir tarafta mutlak, dięerinde baęlı koordinatlar kullanılması, parselin konumunu belirleyen poligonun bir tarafta vekt r dięerinde raster olarak g sterilmesi, parsel numaralarının bir tarafta STRING dięerinde INTEGER olarak kodlanması, parsel alanlarının

bir tarafta *m2* , diğerinde *sq. foot* olarak ifade edilmesi, aynı öznitelik için bir tarafta *kullanım*, diğerinde *kullanım_amacı* etiketleri kullanılması verilebilir.



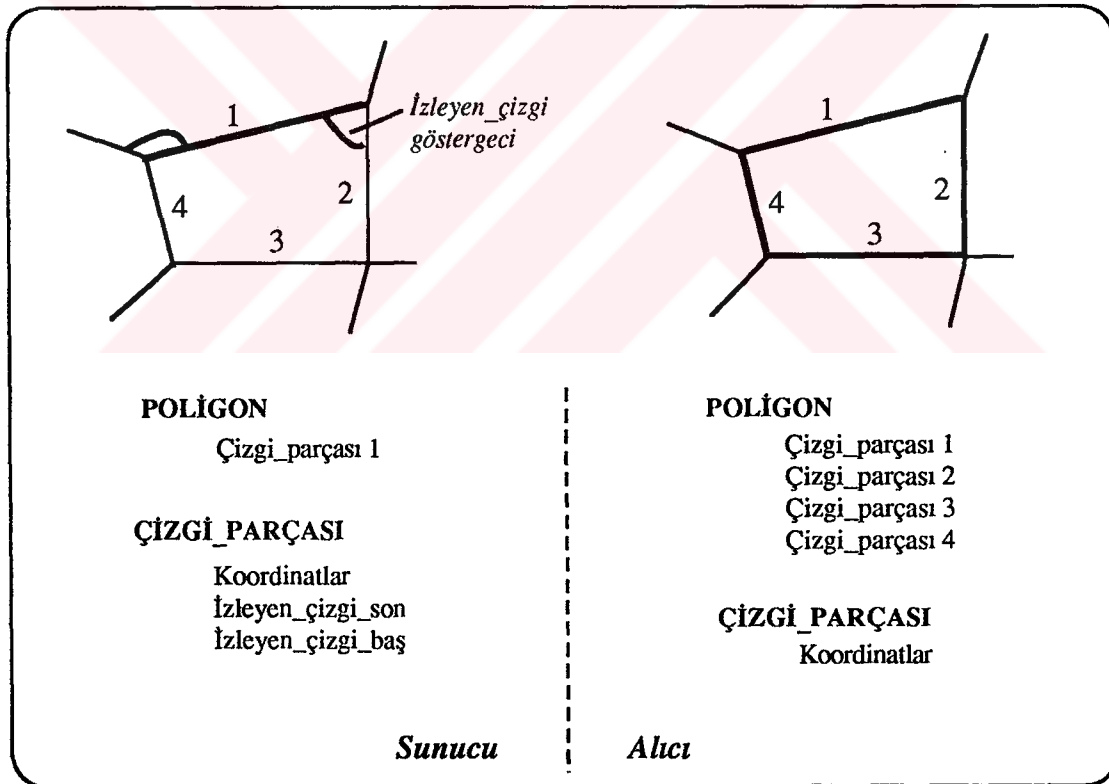
Şekil 12. Veri modeli, şema, ve gerçekleştirim farklılıkları.

İki tarafın veri modellerinin birbirine yakın olması, çeviri işlemini kolaylaştıracaktır. Şema düzeyindeki farklılıkların giderilmesi, veri modeli düzeyindeki farklılıklara göre daha kolaydır [70]. Veri modelleri arasında uyumsuzluk söz konusu olmadığı sürece, farklılıklar çözümlenebilir ve iki sayısal veri gösterimi arasında bir çeviri tanımlanabilir. Aksi durumda, iki tarafın veri değişiminde bulunmaya çalışması zaten anlamsızdır. Söz konusu farklılıkların sınıflandırılması konusunda literatürde görüş birliği sağlanmış değildir [43], [72], [73]. Kapsamlı bir sınıflandırma bu tezin kapsamı dışındadır. Bununla birlikte konumsal veri değişimi kapsamında, çeviriler "karmaşık" ve "basit" olmak üzere iki gruba ayrılabilir.

Aynı bir gerçeğin farklı sistemler tarafından değişik şekillerde algılanması, karmaşık çeviriler gerektirir. Karmaşık çeviriler şema ve/veya veri modeli farklılıklarından dolayı olabilir. Bu tip çeviriler, alıcı veri grubundaki bir veri elemanın değerini elde etmek için, sunucu veri grubundaki bir ya da birden çok veri elemanın değerini girdi olarak alan algoritmalar gerektirir. Bir taraftan diğer tarafa, veri elemanlarının birleştirilme ve ayrıştırılması karmaşık çevirilere örnek olarak verilebilir. Birleştirme ve ayrıştırma, taraflardan birinin nesne-tabanlı, diğerinin kayıt-tabanlı bir konumsal veri modeli kullanması durumunda söz konusu olacaktır. Şekil 12 de, bir tarafın kayıt-, diğerinin nesne-tabanlı olması, veri modeli farklılığına, PARSEL veri tipinin her iki taraftaki tanımları arasındaki

fark ta, şema farklılıklarına örnek olarak gösterilebilir. Yine Şekil 12 ye göre, kayıt-tabanlı taraftan nesne-tabanlı tarafa doğru ilgili PARSEL ve POLİGON verilerinin birleştirilmesi, tersi yönde ise PARSEL nesnesi verilerinin ayrıştırılması söz konusudur.

Farklı topolojik veri yapıları arasındaki çeviriler, karmaşık çevirilere diğer bir örnektir. Örnek olarak, sunucu tarafında poligon dosyasında her poligon için yalnızca bir çizgi parçası tutulabilir ve poligon sınırını oluşturan diğer çizgi parçaları istendiğinde elde edilebilir. Alıcı tarafında ise her poligon için poligon sınırını oluşturan tüm çizgi parçaları bulunabilir (Şekil 13). Bu durumda sunucu tarafından alıcı tarafına geçişte fazla problem yoktur; Her çizgi parçası, baş ve sonundan kendisini izleyen çizgi parçalarına, örneğin Şekil 13 teki gibi, bir göstergeç bulundurduğu için, poligon sınırını oluşturan diğer çizgi parçaları elde edilebilir. Ters yöndeki çeviri daha zordur. Çünkü, alıcı tarafında çizgi parçaları bu tür göstergeçler taşımadığından, sözkonusu çeviri için topolojinin yeniden kurulması ve "izleyen_çizgi" göstergeçlerinin oluşturulması gerekir.



Şekil 13. Sunucu ve alıcı tarafında farklı topolojik veri yapıları.

"Basit çeviriler", tek bir veri elemanı bazında gerçekleştirilebilmeleri bağlamında basittir. Çeviri algoritmaları, alıcı tarafındaki bir veri elemanın değerini elde etmek için, sunucu veri grubundaki tek bir elemanın değerini girdi olarak alır. Tip ve birim farklılıkları bu gruba girer. Örneğin, Şekil 13 te parsel numaralarının bir tarafta INTEGER, diğer tarafta

STRING olarak gösterim edilmesi, parsel alanlarının bir tarafta *m2* , diğerinde *sq. foot* olarak ifade edilmesi verilebilir. Basit çevirelere başka bir örnek olarak, farklı projeksiyon sistemlerindeki koordinatlar arasındaki dönüşümler verilebilir.

Çeviriler sunucu ya da alıcı tarafında gerçekleştirilebilir. Bazı durumlarda çevirinin transferden önce sunucuda ya da transfer anında ("on the fly") yapılarak, verinin alıcı formatında transfer edilmesi gerekebilir. Örneğin, alıcı sistemin topoloji oluşturma programının olmaması fakat topolojik veriye gereksinim duyması durumunda, topolojinin alıcıda kurulması yaklaşımı çalışmayacaktır. Ya da alıcı sistemin "real-time" bir sistem olması durumunda, oldukça zaman alıcı bir işlem olan topoloji oluşturma uygun olmayabilir. Birleştirme/ayırıştırma durumu için de benzer şeyler söylenebilir; Alıcı sistem karmaşık nesneleri kullanmasına karşın, onları oluşturmak için gerekli araçlara sahip olmayabilir.

Konumsal veri değişiminde çeviri işleminden geçen veri, transferden önce ve sonra birbirinin tersi iki işleme tabi tutulur. Transferden önceki kodlama, transferden sonraki ise kod çözümleme işlemidir. Kodlama, belirli bir soyut sözdizimine uyan verinin, belirli kodlama kurallarına göre, *bit/Byte* lardan oluşan bir veri dizisine dönüştürülmesidir. Kod çözümleme ise, kodlanmış veri dizisinin, yine kodlama kurallarına göre yorumlanarak, içerdiği anlamın çözülmesidir. "Soyut sözdizim" [74], kavramsal şemaya eşdeğerdir. [74], soyut sözdizime ek olarak bir de "somut sözdizim" kavramını getirmiştir. Somut sözdizim de yukarıda anılan "kodlama kuralları" na işaret etmektedir.

Verinin soyut sözdizime göre tanımlanmasında üç temel özellik söz konusudur:

1. Bir veri elemanı belirli bir sırada diğer veri elemanlarından oluşabilir. (Bu ASN.1 (Abstract Syntax Notation) [74] de SEQUENCE yapısı olarak tanımlanır):

Harita, sırasıyla bir *dosya başlığı* ve *konumsal detay* lar içerir.

Belirli sıradaki veri elemanları aynı tipte olabilir (ASN.1 de SEQUENCEOF):

Çizgi , *nokta* lardan oluşur.

2. Bir veri elemanı diğer veri elemanlarından oluşabilir; Veri elemanlarının sırası önemli değildir (ASN.1 de SET):

Harita , *nokta* ve *çizgi detay* lardan oluşur. (Konumsal detayların belirli bir sırada olması gerekmez).

Veri elemanları aynı tipte olabilir (ASN.1 de SETOF):

Harita , *konumsal detay* lardan oluşur.

3. Bir veri elemanı bir dizi olası veri tipinden birine tabi olabilir ya da bir dizi olası değerden birini alabilir (ASN.1 de CHOICE):

Konumsal detaylar , *Nokta* ya da *Çizgi* detaylar olabilir.

Harita projeksiyonu , *UTM* , *Mercator* , ya da *Polyconic* ten biri olabilir.

Alıcının, transfer ettiği veriyi yorumlayarak anlayabilmesi için, kodlanmış veri dizisinin bir şekilde anlamlandırılması gerekir. Bunun için çeşitli kodlama teknikleri ve standartları mevcuttur. Genel olarak, iki teknik kullanılır. Bunlardan birincisi "etiket" kullanarak anlamlandırmadır. İkincisinde ise, soyut sözdizimine göre kodlama yapılır. Bu durumda, alıcının transfer ettiği veriyi anlayabilmesi için veri dizisinin içerdiği etiketleri tanınması ya da kodlamanın sözdizimini bilmesi gerekir.

Örnek olarak, aşağıdaki ASN.1 tanımlaması alınsın.

A ::= SEQUENCE { B,C,D }

B ::= SEQUENCEOF { e }

C ::= CHOICE { f,g,h }

D ::= SET {
 i
 j
 K *seçmeli*
 l *seçmeli*
 m }

K ::= SET { n, o, p }

Bu soyut sözdizime göre, belirli bir transfer veri grubunda "C" için, belirtilen üç elemandan yalnızca biri bulunabilir, seçmeli elemanlar bulunmayabilir. Bu durumda bu soyut sözdizim için, aşağıda yalnızca ikisi verilen çeşitli kodlamalar söz konusu olabilir:

1. { {"e1","e2"}, {"g"}, {"i","j"}, {"n","o", "p"}, "m" }
2. { {"e1","e2"}, {"g"}, {"j","i","l","m" } }

Burada her iki kodlamada da, "B" nin "e1", "e2" değerlerini aldığı varsayılmıştır. Alıcı, kodlamanın sözdizim ağacını bilmesi durumunda bu değerlerin "B" ye ait değerler olduğunu anlayabilir. Ancak "C" için, "g" nin hangi elemana ait değer olduğunu, benzer şekilde "m" nin, l nin mi yoksa m nin mi değeri olduğunu çözemez. O nedenle bu belirsizliklerin etiket kullanarak çözümlenmesi gerekir. Veri elemanlarının isimleri etiket olarak kabul edilirse, aşağıdaki kodlama alıcının veri dizisini tek anlamlı olarak çözümlemesi için yeterli olacaktır:

$$1. \quad \{B\{ "e1", "e2" \}, C\{ "g" \}, D\{ "i", "j" \}, K\{ "n", "o", "p" \}, m\{ "m" \} \}$$

Burada **B, C, g, D, i, j, K, m** etiketlerdir. Etiketli kodlamanın avantajı, yalnızca transfer veri grubunda yer alan veri elemanlarının kodlanmasıdır; Yukarıdaki örnekte *l* ye ait bir etiketin olmaması, *l* nin transfer veri grubunda yer almadığını belirtir. Buna karşılık, etiketler çoğunlukla yukarıdaki örnekteki gibi tek karakter uzunluğunda değildirler. Etiketlerin uzun olması ve çok sık kullanılması transfer veri grubunun boyutunu artıracığı için arzu edilmez.

Etiketsiz kodlamada etiket kullanılmaz, kodlama ve kod çözümü soyut sözdizimine göre yapılır. Diğer bir ifadeyle, bir veri elemanının anlamı, veri dizisi içerisindeki pozisyonundan çıkarılır. Bunun için alıcının transfer veri grubunun dayandığı soyut sözdizimini bilmesi gerekir. Yukarıdaki örnekte alıcı sözdizimini biliyorsa, "e1" ve "e2" nin *B* ye ait değerler olduğunu çözecektir. Etiket kullanmama, kayıt alanı gereksinimini azaltacaktır. Ancak, etiketli kodlamada bir veri elemanına ait etiketin bulunmaması, o veri elemanının transfer veri grubunda bulunmadığı anlamına gelirken, etiketsiz kodlamada bu durumun bir şekilde belirtilmesi gerekir. Yukarıda "2" nolu kodlamada, *K* ya ait değerlerin transfer veri grubunda bulunmadığı ancak, *K* nin elemanları için "0" ya da "boş" anlamında başka bir işaret kullanılarak belirtilebilir. Bu da aynen etiketler gibi, kayıt alanı tüketimine neden olacaktır. Transfer veri grubunun kodlanmasında bir diğer seçenek, etiketli ve etiketsiz kodlamanın birlikte kullanılmasıdır. Örneğin, yukarıdaki "1" nolu etiketli kodlamada, *B* ve *K* nin elemanları için etiketlere gerek yoktur. Çünkü *B* aynı tipte elemanlardan oluşmaktadır. *K* ise, sözdizim ağacıyla belirli bir sıradaki elemanlardan oluşmakta ve seçmeli eleman bulunmamaktadır.

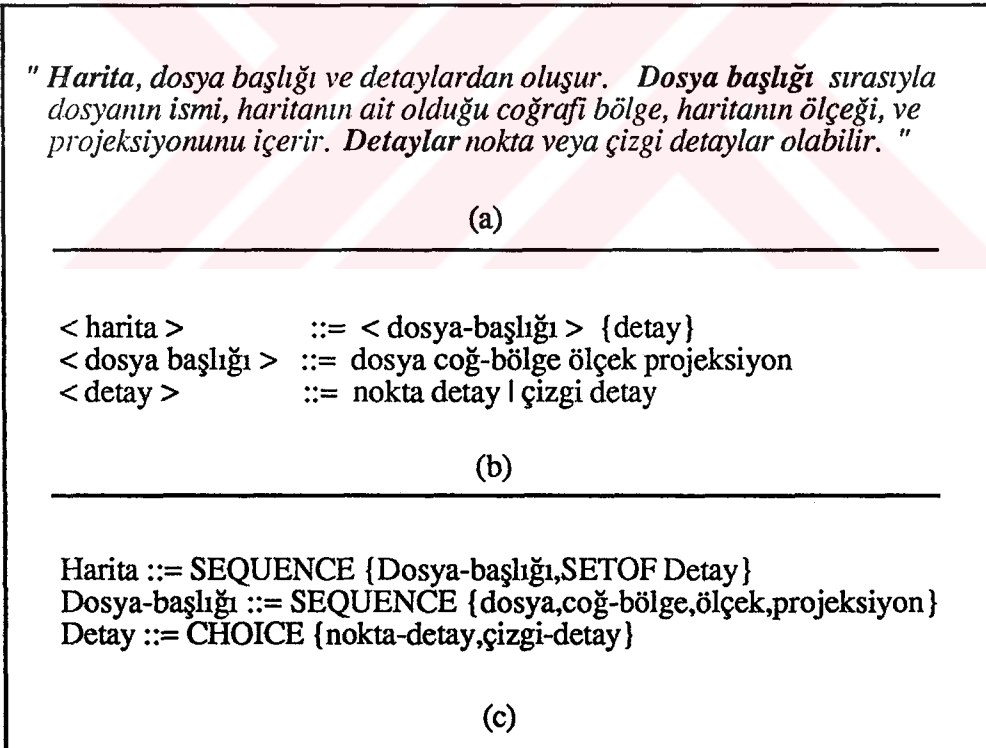
Öte yandan, veri ASCII ya da ikili gösterim modunda transfer edilebilir. ASCII modunun avantajı, ASCII karakterlerinin farklı bilgisayar sistemleri tarafından tanınabilir—donanımdan bağımsız—ve kullanıcı tarafından da okunabilir olmasıdır. Ancak, bir ASCII dosyası ikili modundaki eşdeğerinden daha fazla yer tutar. Örneğin, 255 gibi bir sayının ikili modda saklanması için gerekli kayıt alanı, ASCII modundakinin sadece üçte biri kadardır. Bu bakımdan, daha ziyade sayısal değerlerden oluşan veri için ikili gösterim ASCII gösterimden daha uygundur. Bununla birlikte, değişik bilgisayar sistemlerinin farklı ikili gösterim teknikleri kullanmaları [67] ve ikili dosyaların doğrudan okunabilir ya da yazdırılabilir olmamaları, ikili modu konumsal veri transferinde daha az tercih edilir kılmaktadır.

Verinin kodlanmasında temel ilkelerden biri, olabildiğince kısa ve özlü bir kodlama sağlamaktır. Bu, özellikle çevrim-içi veri transferinde transferin hızı açısından çok önemlidir. Örneğin koordinatların bağıl olarak kodlanması çok sayıda koordinat içeren

detaylar (örn. düzensiz eğri çizgiler) için önemli oranda bir tasarruf sağlayacaktır. Bu bakımdan, etiketle ve etiketsiz kodlama gibi, ASCII ve ikili kodlama da birlikte kullanılabilir. Örneğin koordinatlar ikili, diğer veri ASCII olarak kodlanabilir.

1.3.1.1.1 Kodlama Standartları

Veri değişiminde transfer veri grubunun kodlanması, ISO OSI modelinin sunuş düzeyi kapsamındadır. Verinin belli bir formda sunulması ile ilgili olarak [74], yukarıda kısaca tanımlanan soyut ve somut sözdizim kavramlarını getirmiştir [75]. [74] soyut sözdizimi, "...bir uygulama ile ilgili veri tiplerinin, sayısal gösterimden bağımsız olarak, resmi bir tanımlaması..." olarak tanımlamaktadır. Buna göre, soyut sözdizim kavramsal şemaya eşdeğerdir. Soyut sözdizimdeki bir tanımlamanın, tek anlamlı olması gerekir [76]. Aksi takdirde, sözkonusu tanımlama farklı kullanıcılar tarafından farklı şekillerde yorumlanabilir. Bunun için, "Resmi Tanımlama Dilleri (RTD)" olarak ta anılan soyut sözdizim standartları bulunmaktadır. Bunlardan en çok kullanılan RTD ler, BNF (Backus-Naur Form) [77] ve ASN.1 (Abstract Syntax Notation, ISO 8824) [74] dır.



Şekil 14. Türkçe, BNF, ve ASN.1 dillerinde bir harita tanımı.

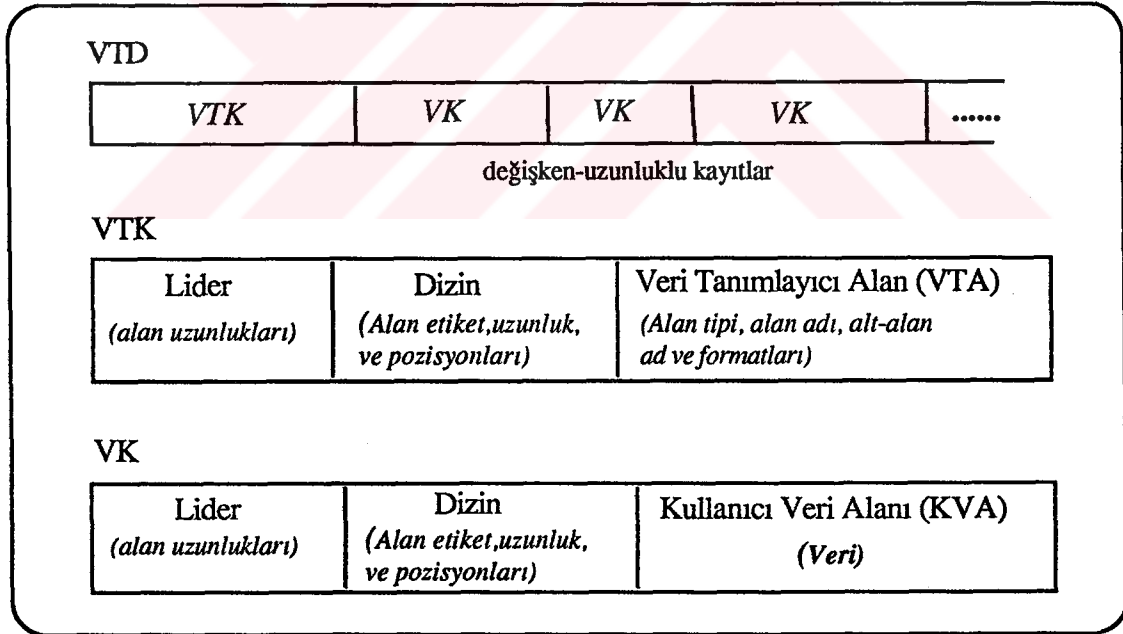
Her RTD nin kendine özgü işaretleri ve sözdizimi vardır. Örneğin BNF "<veri elemanı>" gösterimi, veri elemanının başka bir yerde, daha ayrıntılı olarak açıklanmakta olduğunu gösterir. Aynı şey ASN.1 de, veri elemanının ilk harfinin büyük harf olarak

yazılmasıyla ifade edilir. Şekil 14, bir harita tanımını Türkçe (a), BNF (b) ve ASN.1'de (c) göstermektedir.

Belirli bir soyut sözdizimine uyan verinin fiziksel olarak kodlanması, somut sözdizime göre yapılır. Çeşitli somut sözdizim standartları bulunmaktadır. Ancak burada yalnızca en yaygın kabul görmüş iki kodlama standardı, ISO 8211 [78] ve ASN.1 kodlama kuralları [74] özetle tanıtılacaktır.

1.3.1.1.1.1 ISO 8211

Bir ISO 8211 dosyası, veri ile birlikte verinin tanımını da içerir. Bu nedenle bir ISO 8211 dosyası bazen "veri tanımlayıcı dosya (VTD)" olarak anılır. Her ISO 8211 dosyası, bir Veri Tanımlayıcı Kayıt (VTK) ve onu izleyen bir ya da daha çok veri kayıtları (VK) dan oluşur. Kayıtlar alanlardan, alanlar da alt alanlardan oluşur. Gerek VTK ve gerekse VK lar kayıta yer alan veriyi tanımlayıcı bilgiler ve verinin okunabilmesi için gerekli bilgiler (alanlarda yer alan veri uzunlukları, alanların başlangıç adres ve formatı gibi) içerir. Şekil 15, ISO 8211 dosya yapısını göstermektedir.



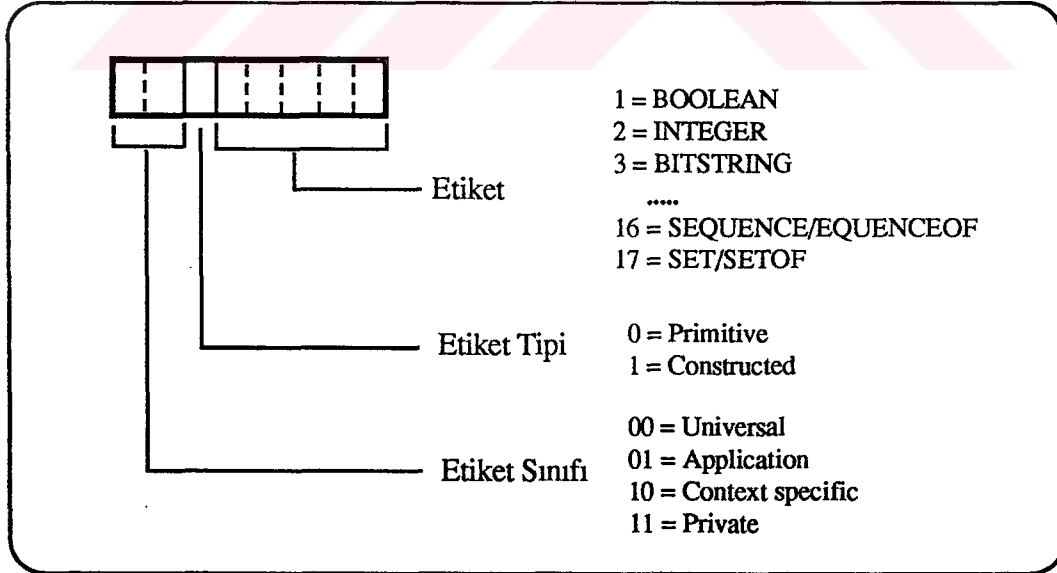
Şekil 15. ISO 8211 (VTD) Dosya yapısı.

ISO 8211'de, etiketli kodlama uygulanır ve ASCII veya ikili modda kodlama yapılabilir. ISO 8211'de, Etiketlerin anlamları VTA'da bulunur. Böylece, VK'da karşılaşılan bir etiketin anlamı, VTK ya bakılarak elde edilir. Ancak bunun için etiket değerinin hem VTK'da hem de VK'da bulunması kodlama için gerekli kayıt alanını önemli ölçüde artırır.

Daha genel bir ifadeyle, ISO 8211'de veri tanımlamanın oldukça detaylı bir şekilde yapılması, veri dosyasını önemli ölçüde genişletir. Bu hem kodlama ve hem de kod çözümlemede verimsizliğe yol açacaktır. Ayrıca ISO 8211, kökleri 1970 lere dayanan bir standart olarak, özellikle çevrim-dışı veri transferi için daha uygundur [70], [79].

1.3.1.1.1.2 ASN.1 Kodlama Kuralları

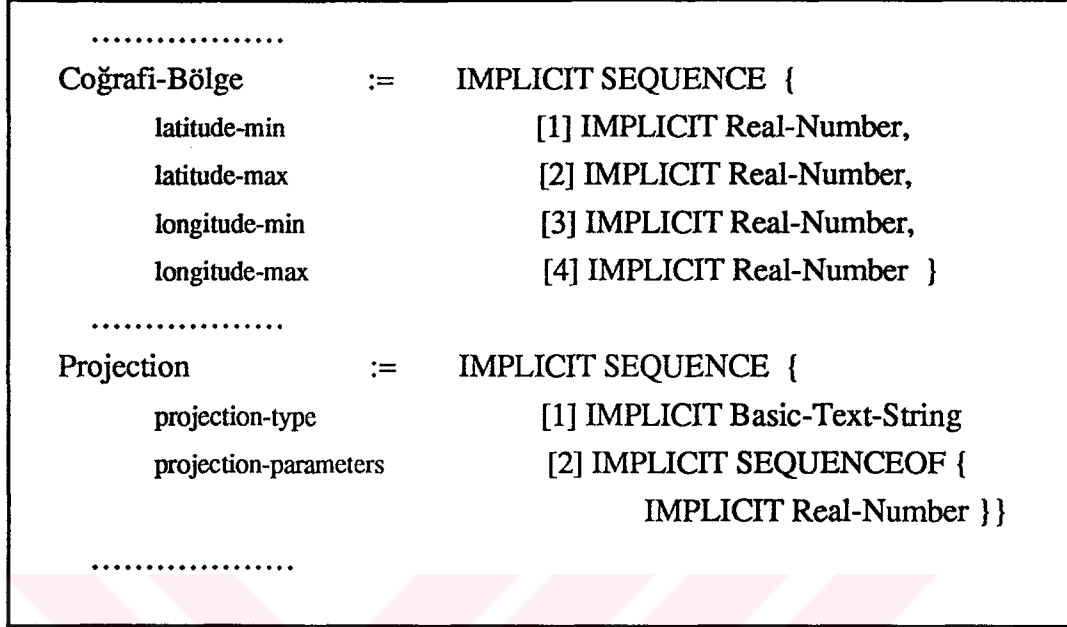
ASN.1'de de etiketli kodlama uygulanır. ASN.1, dört çeşit etiket tanımlar. Bunlar "evrensel", "uygulama", "kapsam özellikli", ve "özel" etiketlerdir. Evrensel etiketler, ASN.1'in kullanıldığı bütün kodlamalar için geçerlidir ve aynı anlamı taşır. Örneğin tamsayılar için "2", SET ve SETOF (bkz. Şekil 14) yapıları için de, "17" evrensel etiketleri kullanılır. Uygulama etiketleri, belirli bir transfer için tek anlamlıdır. Yani bir uygulama etiketinin iki ayrı transfer veri grubundaki anlamları farklı olabilir. Kapsam özellikli etiketler ise, belirli bir transfer veri grubunun belirli bir alt grubu için tek anlamlıdır. Örneğin, bir karayolunun çeşitli öznitelikleri için kapsam özellikli etiketler kullanılabilir. ASN.1'de kapsam özellikli etiketler köşeli parantez içinde tamsayılarla ("[1]" gibi) gösterilir. Kod çözücünün etiketleri yorumlayabilmesi için, etiket tipini bilmesi gerekir. Bunun için ASN.1'de etiketler, "octet" adı verilen *Byte* larda tutulur. Buna göre, 1 *Byte* üç alana ayrılmıştır (Şekil 16). İki bitlik ilk alan etiket tipini, ikinci alan verinin ASN.1'de tanımlı iki tipten hangi gruba girdiğini, üçüncü alan ise etiketi gösterir.



Şekil 16. ASN.1 de etiketler.

ASN.1 de kodlamaya bir örnek olarak, Şekil 14 teki harita tanımlamasına dönülsün. Şekil 14 (c) deki tanım, Kanada'nın MACDIF formatının harita tanımıdır

(genelleştirilmiştir). MACDIF [80]'te, haritanın ait olduğu coğrafi bölgenin ve harita projeksiyonunun ASN.1 deki tanımlaması, Şekil 17 deki gibidir.



Şekil 17. MACDIF formatının, ASN.1 deki coğrafi-bölge ve projeksiyon tanımı.

ASN.1, özlü kodlamayı desteklemektedir. Bunun için, kodlayıcının gerek olmayan durumlarda etiket üretmesini engellemek amacıyla, "IMPLICIT" deyimi kullanılır. Bu durumda kod çözücünün, soyut sözdizimini bilmesi gerekir. Buna göre Şekil 17 deki IMPLICIT terimleri burada [1]...[4] gibi kapsam özellikli etiketlerin üretimini ve transfer veri grubu içinde kodlanmasını engelleyecektir. Soyut sözdizimini bilen bir kod çözücü, projeksiyon verisinin projeksiyon-tipi ve onu izleyen projeksiyon-parametrelerinden oluştuğunu ve bu verilerin mutlaka transfer veri grubunda bulunması gerektiğini—seçmeli olmadıklarını—bileceği için, ayrıca etiket kullanmaya gerek kalmamaktadır. ISO 8211'den daha sonra geliştirilmiş bir standart olarak, ASN.1 çevrim-içi veri transferi için daha uygundur .

1.3.1.2 İletişim

Konumsal veri değişiminde sözkonusu olan, bir tarafın (alıcı) diğer taraftan (sunucu) veri istemesi, buna göre gerekli verinin derlenmesi, ve bir şekilde transfer edilmesidir. İki tarafın aynı veriyi çok farklı şekillerde gösterim etmeleri bir yana, burada en önemli sorun, iki taraf arasında iyi bir iletişim kurulmasıdır. Bu iletişim sayesinde, daha işin en başında alıcı, sunucunun gerekli veriyi sağlayıp sağlayamayacağını ya da ne oranda sağlayacağını belirleyecek, ve "neyi", "nasıl" istediğini belirtecektir. Bu iletişim her iki tarafın

anlayabileceği ortak bir dil kullanmalıdır. Sözkonusu iletişim için, konumsal veri değişim ortamının, kullanıcılara gerekli "arabirim" ve "servisler"i sağlaması gerekir.

1.3.1.2.1 Arabirimler

İdeal bir veri değişim ortamında veri değişimi, veri gruplarını grafik olarak görme, tarama, sorgulama ve ardından istenen alt grupları ya da nesneleri transfer etmeye izin veren "Grafik Kullanıcı Arabirimleri (GKA)" ile gerçekleştirilmelidir. Böyle bir GKA desteğinde alıcı, ilgili sunucu verisini tarayıp sorgulayarak, yalnızca gereksinim duyduğu veriyi transfer eder. Diğer bir ifadeyle, "seçmeli transfer" ya da "nesne-tabanlı transfer" olanaklı olmalıdır.

Konumsal veri değişiminin bugün pratikte uygulanışı ele alındığında, seçmeli transfer en önemli eksikliklerden biri olarak ortaya çıkmaktadır. Çünkü konumsal veri değişimi bugün hala çoğunlukla "batch" modunda, bütün bir dosyanın transferi şeklinde gerçekleştirilmektedir. Alıcı sistemin gereksinim duymadığı veriyi ithal etmesi, bu veriye kodlama, transfer, kod çözümü, ve muhtemelen gereksiz veri olarak ayırma işlemlerinin uygulanmasını gerektirecek ve böylece transfer zamanını artıracak ve sonuçta kaynak israfına neden olacaktır. Modern CBS lerin buna tahammülü yoktur. Seçmeli transfer için, "seçim araçları" olarak nitelendirilebilecek görme, tarama, ve sorgulama araçlarına ya da bu fonksiyonları sunan GKA lara gereksinim vardır.

Seçim araçları, transfer edilecek detayların tek tek ya da grup olarak belirtilmesine olanak tanınmalıdır. "5555" nolu parselin ya da "TEM Otoyolu" nun belirtilmesi tek tek detay belirtmeye, "Et Balık Kurumuna ait bütün parseller" in, "üç şeritli yollar" ın ya da belirli bir coğrafi bölge içerisindeki detayların belirtilmesi de, grup halinde detay belirtmeye örnektir. Dolayısıyla, transfer edilecek verinin seçimi, örneklerdeki gibi özniteliklere ya da konumsal kapsama göre veya grafik görüntü üzerinden yapılabilmelidir. Bunun için bir sorgulama dili ya da nesneleri anılan şekilde belirtmeye izin veren bir sözdizim ve görme ve tarama araçları gereklidir.

Görme ve tarama araçları transfer edilecek verinin grafik olarak görüntülenmesi ve incelenmesi için gereklidir. Görme araçları ile örneğin zoom yapılabilecek, görüntüleme renkleri değiştirilebilecektir. Tarama araçları ile örneğin bir nesnenin öznitelikleri diğer özellikleri (örn. verinin doğruluğu, kaynağı) listelenebilecek, hangi ölçüte (yer, öznitelik) göre tarama yapılacağı belirtilebilecektir [40].

Konumsal veri değişimi alanındaki araştırmalar bugüne kadar hep değişim formatı tasarımı üzerinde yoğunlaşmış, bu formatların nasıl uygulanacağı konusu çoğunlukla

gözardı edilmiştir. Bu nedenle geleneksel değişim formatları, yalnızca formatın soyut ve somut tanımını kapsamakta, formatın uygulanışı—çevirici geliştirme vs.—tamamen gerçekleştirimcilere bırakılmaktadır [81]. Bunun sonucu olarak, konumsal veri değişiminin bugün pratikte uygulanışına bakıldığında, yukarıda anılan GKA ların eksiksikliği göze batmaktadır [15], [40]. "DCW (Digital Chart of the World)" ürünü, sözkonusu yetenekleri içeren bir arabirim ile desteklenmektedir. DCW, bir değişim formatı değildir. 1:1 000 000 ölçeğinde, Dünya genelinde çeşitli veriler içeren bir veri kaynağıdır.

1.3.1.2.2 Veri Transferi

Veri değişiminde verinin sunucudan alıcıya fiziksel olarak transferi, çevrim-dışı ya da çevrim-içi bir tarzda gerçekleştirilebilir. Geleneksel yöntem olan çevrim-dışı veri transferinde, sunucu verisi manyetik teyp, optik disk (CD-ROM), disket vs. gibi kayıt ortamları üzerine kaydedilerek alıcıya gönderilir. Geleneksel kayıt ortamı olan manyetik teypler, bugün artık yerlerini hızla CD-ROM lara bırakmakta ve daha ziyade arşivlemede kullanılmaktadır. CD-ROM lar gerek yazılım ve gerekse veri dağıtımında (Örn. hemen bütün KVYS yazılımları, DCW ürünü) yaygın olarak kullanılmaktadır.

CD-ROM ların manyetik teyplere tercih edilmesindeki en önemli etkenlerden biri, CD-ROM larda erişim hızının daha yüksek oluşudur. Çünkü CD-ROM lar doğrudan erişimi desteklemekte, manyetik teypler ise yalnızca sıralı erişime izin vermektedir. Diğer etken, manyetik teyplerde kullanılan kayıt formatının donanım bağımlı olabilmesidir. Oysa CD-ROM larda kayıt formatı için standartlar mevcuttur [70].

Çevrim-içi veri transferinde, transfer doğrudan iki taraf arasındaki iletişim hattı üzerinden gerçekleşir. Hızla değişen gereksinimler, iletişim teknolojisindeki son gelişmelerin desteğiyle, farklı sistemlerin federe bir yapı içerisinde çalışmasını zorunlu kılmış ve çevrim-içi veri transferinin önemini artırmıştır. Bugün artık, çeşitli sistemler mesafe tanımaksızın birbirine fiziksel olarak bağlanabilmektedir. İletişim hattı kablolu ağları, telefon hatlarını, ve uydu kanallarını kullanabilir (bkz. kısım 1.4). Çevrim-içi veri transferinde, hem veri transferi CD-ROM ve teyplere göre çok daha hızlı, hem de iletişim servisleri çok daha çeşitlidir.

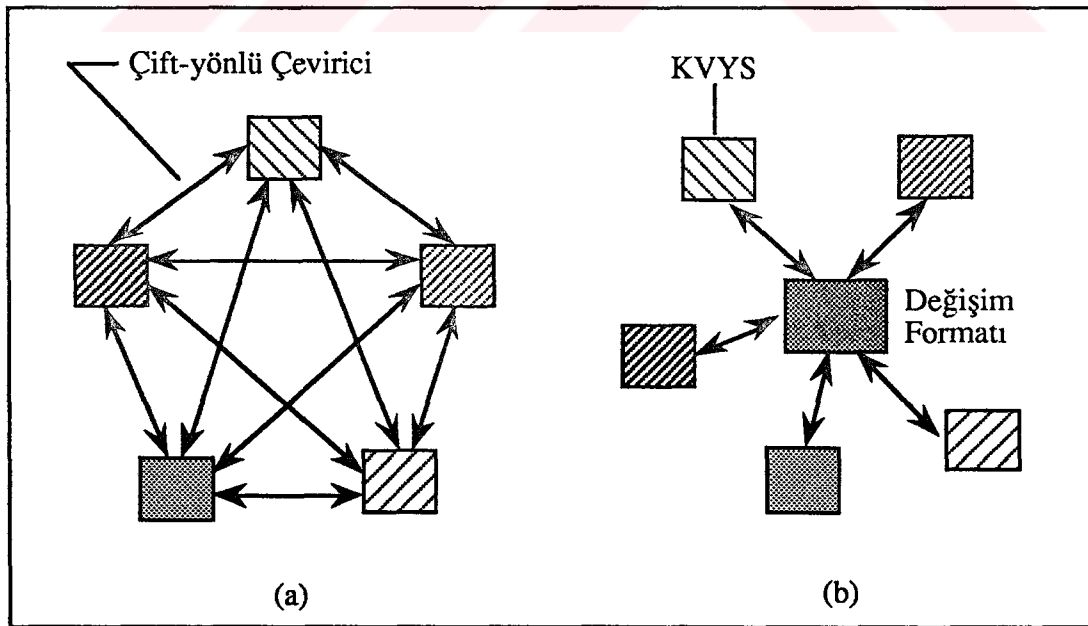
1.3.1.2.3 İletişim servisleri

İletişim servisleri ISO OSI modeli [71] nin, 1 den 5 e kadar olan düzeylerine işaret eder. İletişim servisleri, çevrim-içi ve çevrim-dışı servisler olmak üzere iki ayrı grupta düşünülebilir [29]. Çevrim dışı servisler, yukarıda söz edildiği gibi çevrim dışı tarzda

yazılım ve veri transferi içindir. Çevrim içi servisler ise, bilgisayar sistemlerini dünya genelinde birbirine bağlayan iletişim ağlarının sunduğu servislerdir. Bunlar "diyalog servisleri", "elektronik mail", ve "dosya transferi" servisleridir. Bu servisler yardımıyla, Dünya'nın bir ucundan diğerine iki sistem birbiriyle "konuşabilir", birbirine mesaj gönderebilir ve birbirinden dosya transfer edebilirler. Diyalog servislerine basit bir örnek olarak, TCP/IP (Transfer Control Protocol/Internet Protocol) deki "talk" komutu verilebilir. Yine TCP/IP deki "ftp" (File Transfer Protocol) servisi ise dosya transferi servislerine örnektir. Ancak bu servisler konumsal veriye yönelik olmayan genel amaçlı servislerdir. O nedenle bu servislerin, konumsal veri için genişletilmesi ya da yeniden tanımlanması gerekir. Konumsal veri değişimi için oldukça genel fonksiyonlar sunmakta olan bu servisler, yukarıda sözedilen arabirim ve araçlar için gerekli alt yapıyı oluşturabilirler.

1.3.2 Konumsal Veri Değişim Yöntemleri

Genel olarak, konumsal veri değişimi iki farklı şekilde gerçekleştirilebilir. Bunlar, "doğrudan" ve "dolaylı" yöntemlerdir (Şekil 18). Doğrudan yöntemde (Şekil 18a) sunucu ve alıcı arasında "doğrudan çevirici" ler kullanılır. Sunucu ve alıcı formatları arasında doğrudan bir çeviri söz konusudur. Sunucu verisi, sunucu ya da alıcı formatında alıcıya gönderilebilir. Dolaylı yöntemde (Şekil 18b) ise, "evrensel çevirici" ler kullanılır. Bu durumda, çeviri işlemi evrensel çeviricinin kullandığı değişim formatı üzerinden, dolaylı olarak yapılır. Aşağıda bu yöntemler hakkında ayrıntılı bilgi verilmektedir.



Şekil 18. Konumsal veri değişim yöntemleri.

Literatürde [14], [81] üçüncü bir yöntem olarak gösterilen "ring" yöntemi, buradaki sınıflamaya göre dolaylı yöntem sınıfına girdiği için burada ayrı bir yöntem olarak ele alınmamıştır. Çünkü ring yönteminde de, değişim formatı niteliğinde bir ara format söz konusudur. İki farklı format arasında her ikisi için de ortak üçüncü bir format üzerinden transfer yapılabilmesi durumunda, ikiden çok sistem arasında ring yöntemi uygulanabilir.

1.3.2.1 Dolaylı Yöntem

Dolaylı yöntemde iki format arasındaki çeviri bir "ara format" üzerinden gerçekleşir. Sunucu verisi önce ara formata dönüştürülür, alıcı ara formatta veriyi transfer eder ve kendi formatına dönüştürür. Bu örneğin, Fransızca'dan Türkçe'ye bir çeviri yaparken, Fransızca'dan önce İngilizce'ye ve oradan da Türkçe'ye geçmeye benzetilebilir. Burada İngilizce "ara format" a eşdeğer bir görev üstlenmektedir. Buradaki "ara format" çoğunlukla "değişim formatı", bazen de "evrensel format", ya da "nötür format" olarak anılır [14], [82], [83]. Yayımlanmış ve yaygın bir kullanım alanı bulmuş bir değişim formatı "değişim standardı" olarak adlandırılır [81].

Dolaylı yöntem, çok sayıda farklı taraf içeren organizasyonlar için (örn. dağıtılmış sistemler), gereksinim duyulan çevirici sayısı ve organizasyona yeni tarafların eklenebilmesi açısından, doğrudan yöntemle göre çok daha pratik bir çözüm sunmaktadır. Şöyle ki, N sayıda tarafın bu yöntemle göre veri değişiminde bulunabilmesi için, çeviricilerin çift yönlü olduğu, yani değişim formatı ile diğer format arasında birinden diğerine çeviri yapabildiği varsayımıyla, sadece N sayıda çeviriciye gereksinim vardır (Şekil 18b). Çünkü her bir taraf için, yalnızca değişim formatına yönelik bir çevirici yeterlidir. Herhangi bir sistemin iç formatını değiştirmesi, yalnızca ilgili çeviriciyi etkileyecektir. Benzer şekilde organizasyona yeni bir tarafın eklenmesi durumunda, yalnızca yeni taraf için çevirici geliştirmek gerekecek diğer taraflar bu değişiklikten etkilenmeyecektir.

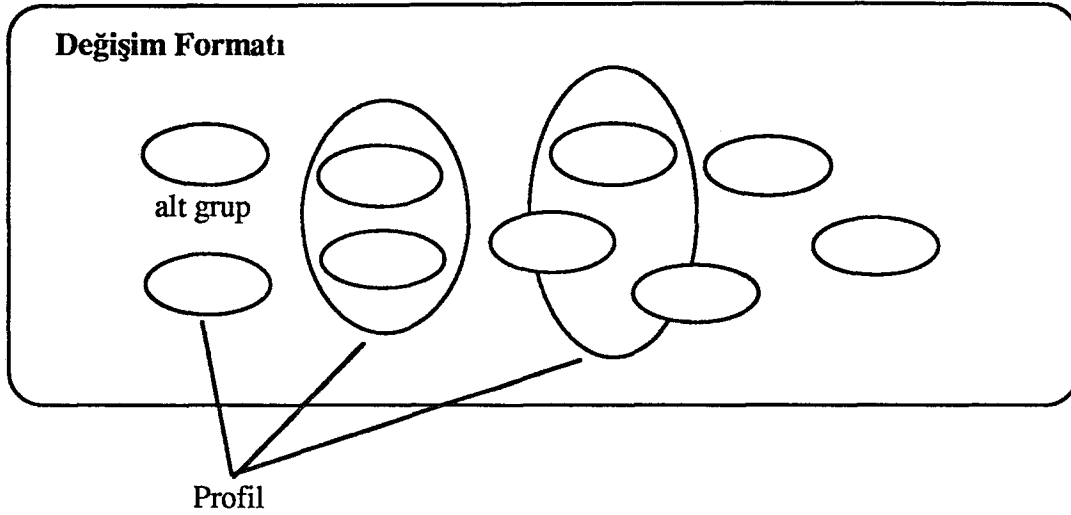
Mümkün olduğunca çok sayıda farklı format arasında eksiksiz ve etkin bir veri değişiminin sağlanabilmesi için, veri değişim formatının bir takım özelliklere sahip olması gerekir. Buna göre, değişim formatı "genel", "esnek", ve "genişletilebilir" olmalı, donanımdan bağımsız, farklı bilgisayar ve iletişim ortamlarında kullanılabilir olmalıdır. Bunlara ek olarak formatın, gerçekleştirimini kolaylaştıracak destekleyici yazılım ve arabirimlerle desteklenmesi, formatın bir "standart" olarak yaygın kabul görmesinde önemli rol oynayacaktır. Değişim formatlarının taşınması gereken özellikler 2. Bölümde ayrıntılı olarak ele alınmaktadır.

Değişim formatının iki format arasında eksiksiz bir çeviri sağlayabilmesi için, her iki formatı dayandıkları soyut ve somut sözdizimleri bazında kapsamaları gerekir. Aksi durumda çeviri eksik olacak ve "bilgi kaybı" oluşacaktır. Örneğin 3 boyutlu koordinatlarla çalışan taraflar arasında, koordinatların yalnızca 2 boyutlu olarak tanımlandığı bir değişim formatı ile veri değişiminde, "z" koordinatı transfer edilemeyecek ve bir bilgi kaybı söz konusu olacaktır. Formatın genelliği, eksiksiz çevirisi yapılabilen format sayısı ile ilişkilidir. Bu bağlamda örneğin, farklı raster ve vektör gösterimleri arasında çeviriler olanaklı olmalıdır. Eksiksiz çevirisi yapılabilen format sayısı arttıkça, değişim formatının kapsamı da genişleyecektir. Konumsal veri yönetimi alanında gerek veri tipleri ve gerekse veri yapılarının çeşitliliği, bunun üstesinden gelebilecek bir değişim formatının çok kapsamlı olmasını gerektirmektedir.

Öte yandan, değişim formatının genel olması formatı özellikli transferler için aşırı kapsamlı ya da hantal kılacaktır. Örneğin yalnızca raster veri içeren bir değişim için, hem raster hem vektör veriye yönelik bir değişim formatının kullanılması verimsizliğe yol açacaktır. O nedenle formatın özellikli değişimler için uyarlanabilir, yani esnek olması gerekir. Örneğin değişik çözünürlüklerde koordinat gösterimi mümkün olmalı, veya belirli bir değişim için yalnızca 2 boyutlu koordinatların geçerli olduğu belirtilebilmelidir.

Gerçekleştirim açısından, kapsamlı bir formatın tek bir çevirici ile gerçekleştirimi, geniş bir çevirici programı gerektirecektir. Geniş bir çevirici, özellikli veri değişimleri için hantal kalacak ve bilgisayar kaynaklarının israfına neden olacaktır. Ayrıca, böyle bir yazılımın geliştirilmesi hatırı sayılır bir zaman alacaktır. Bu nedenlerle, aynı formatın değişik kullanıcı grupları tarafından farklı şekillerde uygulanması kaçınılmaz olmakta ve bu da standartlaştırmanın espirisine ters düşmektedir [82], [83]. Esasen dolaylı yöntemin ikilemi, esneklik ve standartlaşmanın ikisini birden sağlamaya çalışmaktır [83]. Bu iki unsuru dengede tutmak ve kabul edilebilir sürelerde, "çalışır" çeviriciler üretmeyi mümkün kılmak amacıyla, son zamanlarda geliştirilmiş formatlar, formatın "profil" ler vasıtasıyla uygulanmasını öngörmektedirler [22], [84], [85].

Bir değişim formatı profili, formatın özellikli bir alt grubudur (Şekil 19). Genel bir formatın bir profilinin oluşturulması, profil kapsamındaki veri tiplerinin, alabileceği değerlerin, ve ilgili kısıtlayıcıların belirlenmesini içerir. Buna göre bazı profiller formatın birden çok alt grubu ile ilişkili olabilir. Profillere Örnek olarak, hem raster ve hem de vektör veriye yönelik bir değişim formatının, sadece vektör veri değişimine izin veren bir profili, ya da metrik ve topolojik veri değişimine izin veren bir formatın metrik veri profili gibi.



Şekil 19. Profil kavramı.

Bir değişim formatı tasarımında, gelecekteki gereksinimleri tasarım aşamasında kestirmek çok zor, belki de olanaksızdır. Bu nedenle bir değişim formatının zamanla değiştirilmesi ya da genişletilmesi kaçınılmazdır [81]. Önemli olan değişikliklerin kolay ve hızlı bir şekilde yapılabilmesidir. Örneğin yeni veri tipleri kolaylıkla formata eklenebilir olmalı, böylesi değişiklikler mevcut çeviricinin önemli oranda yeniden yazılmasını gerektirmemelidir. Bunun için değişim formatının "genişletilebilir" bir veri modeline sahip olması gerekir.

Gerçekleştirim açısından, değişim formatının donanımdan bağımsız, farklı bilgisayar ve iletişim ortamlarında kullanılabilir olması gerekir. Bir değişim formatı tanımı formatın soyut ve somut tanımlarını içerir [81]. Soyut tanım, formatın şema ve veri modelini soyut sözdizime göre tanımlar. Somut tanım fiziksel kodlama ile ilgilenir. Değişim formatının donanım ve iletişim ortamından bağımsızlığını sağlamak için, son zamanlarda geliştirilmiş formatlar [22], [35], formatın soyut ve somut tanımları arasında bir ayırım yapmakta, ikisini birbirinden bağımsız olarak tanımlamaktadır. Böylece format değişik kodlama yöntemleri ile birlikte kullanılabilir. Gerçekleştirim açısından diğer önemli nokta, değişim formatının soyut ve somut tanım yanında, değişim ortamı detaylarını da kapsamasıdır. Bu bakımdan bir değişim formatının, 1.3.1.2.1 de belirtildiği gibi bir GKA ya da formatın gerçekleştirimini kolaylaştıracak bir takım araçları kapsamında bulundurması önemli bir ayrıcalıktır. Çünkü bu durum bir değişim formatının standartlaşmasını önemli ölçüde etkileyecektir.

Dolaylı yöntemde en yüksek yarar, gerek tek bir organizasyon içerisinde ve gerekse organizasyonlar arası düzeyde değişim formatı olarak tek bir standart format kullanılmasıyla elde edilecektir. Ancak bu ideal durum, teknik ve teknik olmayan gibi iki ayrı grupta düşünülebilecek çeşitli nedenlerden ötürü, bugüne kadar gerçekleşmemiştir. Sonuç olarak bugün, dolaylı yöntem konumsal veri değişiminde en çok kullanılan yöntem olmakla birlikte, çok çeşitli formatlar, değişim formatı olarak kullanılmaktadır. Bunlardan en çok kullanılan bazıları DXF (Drawing Exchange Format) [86], DLG (Digital Line Graph) [87], SIF (Spatial Interchange Format) [88] dir.

Bu durum, dolaylı yöntemle amaçlanan "standartlaşma" nın espirisine ters düşmektedir. Çünkü değişik KVYS ve veri sağlayıcı kuruluşların değişim formatı olarak farklı formatlar kullanmaları, çok değişik kaynaklardan veri transfer etmek durumunda olan organizasyonları, ilgili formatların her biri için gerekli çeviricileri yazılım kapsamında bulundurmak zorunda bırakmıştır. Daha önemlisi çeviricilerin sürekli yenilenme gereksinimidir. Çünkü değişim formatları durağan olmayıp, farklı kullanıcı istekleri nedeniyle zaman içerisinde sürekli değişikliğe uğrarlar [14], [81], [83]. Bu durumda değişim formatının farklı versiyonları arasında uyumluluğun sağlanması gerekir. Diğer bir deyişle bir KVYS, aynı bir değişim formatının farklı versiyonlarında kodlanmış veri gruplarını transfer edebilmelidir. Bu problem bazen "versiyon sürekliliği" olarak anılır [81].

Konumsal veri değişiminin bugünkü uygulanış şeklinde sunucu ve alıcı, çevirisine sahip oldukları, en uygun bir değişim formatı üzerinde anlaşmak durumundadır. Değişim formatının yeterli genellik ve esnekliği sağlayamaması durumunda, değişim formatı üzerinden transferde doğrudan transfere kıyasla bir bilgi kaybı riski ve çevirinin daha zor olması olasılığı vardır.

Standartlaşmanın önündeki temel engel bir değişim standardı üzerinde görüş birliğinin sağlanamamasıdır. Bunda "Standart" olarak yaygın kabul görecektir "iyi" bir değişim formatı tasarımının ve tek tip gerçekleştiriminin teknik açıdan oldukça güç olmasının ve uzunca bir süreç gerektirmesinin rolü çok büyüktür [11], [14], [21], [81], [82]. Örneğin A.B.D.'nin SDTS (Spatial Data Transfer Standart) [22] formatının geliştirilmesi, dokuz yılda ancak tamamlanabilmiştir. Bunun yanında, kapsamlı bir format doğal olarak oldukça karmaşık olacağı için, gerçekleştirimi de belirli bir süreç gerektirecektir. Çünkü böyle bir formatı uygun bir şekilde anlamak ve uygulamak, belirli bir uzmanlık düzeyi gerektirmektedir [40], [89]. Bu durumda, konumsal veri yönetimi gibi çok dinamik bir alanda beklemeye tahammülü olmayan kullanıcılar, yeni bir değişim formatının gerek tasarım ve gerekse gerçekleştirim bakımından uzun süreç gerektirmesi nedeniyle, ya mevcut bir formatı kullanmışlar ya da çok daha kısa bir sürede ihtiyaca cevap veren bir format geliştirmişler ya

da aynı bir formatı farklı şekillerde uygulamışlardır. Sonuçta da, konumsal veri değişiminde bugün yaşanan tablo ortaya çıkmıştır.

Bugüne kadar tek ya da çok az sayıda konumsal veri değişim standardına ulaşılamamasında, KVYS satıcılarının, "pazar politikaları" gereği, yeterli desteği sağlamamalarının rolü çok büyüktür. Bu politikaları belirleyen etkenlerden biri, bir değişim standardı geliştirmenin oldukça pahalı ve riskli bir işlem olması ve standart kabul görse bile bunun uzun vadeli bir yatırım olmasıdır. Dahası bir değişim standardı, satıcının kendinden çok diğer satıcılara yarayabilir. Çünkü, veri değişim problemi olmaması durumunda, kullanıcılar belirli bir sisteme bağlı kalmak yerine diğer sistemlere yönelebilirler. Ayrıca, veri değişiminin kendi başına oluşturduğu bir pazar—çevirici yazılımları ve ilgili bakım hizmetleri—sözkonusudur. Bu nedenlerle, KVYS satıcıları değişim standartları konusunda çekimser davranmaktadırlar [11].

Standart bir değişim formatına ulaşmada diğer önemli sorun kullanıcılar arasında görüş birliğinin sağlanamamasıdır. Buradaki temel problemlerden biri, farklı kullanıcıların formatın mevcut ve gelecekteki ihtiyaçları karşılayacak yeterlikte olup olmadığı üzerinde farklı görüşlere sahip olmalarıdır. Diğer sorun, bir değişim standardının tasarımı ve yaygınlaşmasının oldukça uzun bir süreç gerektirmesidir. Oysa pek çok projenin beklemeye tahammülü yoktur. Nihayet, yeni bir formata geçişin maliyeti yüksektir [11].

Standartlaşma açısından üçüncü önemli sorun, standart geliştirici organizasyonlar arasında birlik sağlanamamasıdır. Bunun sonucunda farklı organizasyonlar, birbirinden bağımsız çalışmalar sonucunda farklı "standart" larla ortaya çıkmaktadır. [11], konumsal veri değişim standardı geliştiricilerinin, yakın zamana kadar ISO OSI modeli çalışmalarını gözardı ettiklerini belirtmektedir. Bunun sonucunda, standartlaştırma çabaları standartları azaltmak yerine, mevcut standartlara yenilerini eklemiştir.

Sonuç olarak, KVYS satıcı/geliştiricileri, KVYS kullanıcıları, ve standart geliştirici organizasyonlar arasında belirli bir değişim formatı üzerinde görüş birliği sağlanamamışsa, bu değişim formatının bir değişim standardı olarak kabul görme şansı, format ne kadar özenle ve uzun zamanda geliştirilmiş olursa olsun, çok azdır. Bu durumda, birtakım yasal düzenlemelerle formatın yaygınlaşmasını zorlamaktan başka çare kalmamaktadır. A.B.D.'nin SDTS formatı için de durum böyle olmuş ve konumsal veri sağlayan devlet kuruluşlarının, Şubat 1994'ten başlayarak SDTS formatında veri sağlamaları zorunluluğu yasal olarak getirilmiştir [82].

Bugüne kadar gerek teknik ve gerekse teknik olmayana nedenlerle çeşitli kesimlerin "kendi" standartlarını geliştirmiş olmaları sonucu, konumsal veri değişimi alanında bugün çok çeşitli "standartlar"ın kullanılıyor olması, standartlaştırma ile amaçların tam tersi bir durumdur. Giriş bölümünde değinildiği üzere, kullanılan standart sayısının en aza indirilmesi amacıyla son bir kaç yıldır, DIGEST, SDTS, SAIF, ve DX-90 [27] gibi başlıca konumsal veri değişim formatları arasında uyum sağlanması çalışmaları gündeme gelmiştir.

1.3.2.2 Doğrudan Yöntem

Doğrudan yöntem ile veri değişiminde bulunacak taraflar, karşı tarafın formatında veri ithal ya da ihraç edebilmelidir. Tarafların ikiden çok olması durumunda, her bir çift için bu kural geçerlidir. Dolayısıyla N sayıda tarafın bu yönteme göre veri değişiminde bulunabilmesi için, çeviricilerin çift yönlü olduğu, yani iki format arasında birinden diğerine çeviri yapabildiği varsayımıyla, $N*(N-1)/2$ çeviriciye gereksinim vardır (Şekil 18a). Dağıtılmış sistemler bağlamında, farklı sistem sayısının artmasıyla gereksinimin duyulan çevirici sayısı da katlanarak artacaktır. Daha da kötüsü, sistemlerden birinin iç formatında birtakım değişiklikler yapması durumunda, (N-1) adet çeviricinin yenilenmesi ya da yeniden yazılması gerekir. Bu bakımdan dolaylı yöntem, doğrudan yöntemle önemli bir üstünlük sağlamaktadır.

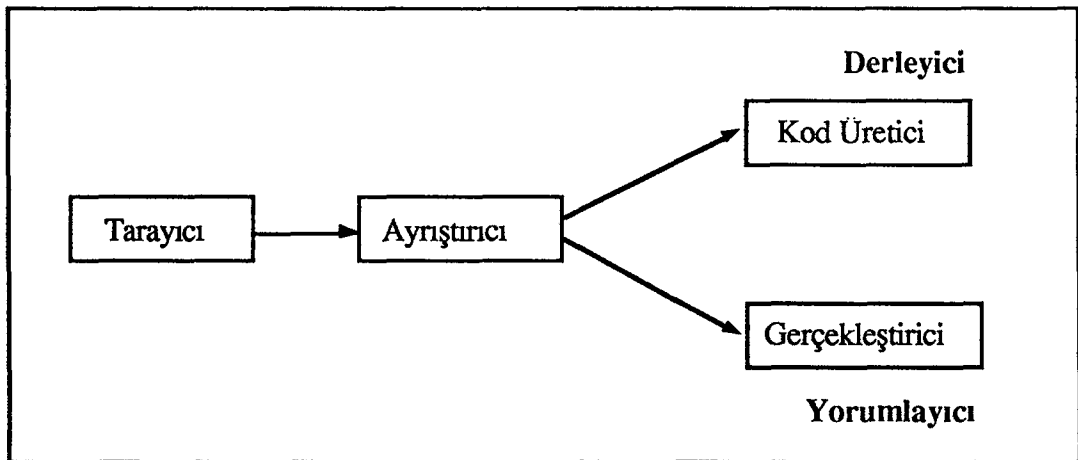
Bununla birlikte, doğrudan yöntemde çeviri yalnızca iki formata yönelik olduğundan, "çeviri kalitesi" yüksektir. Diğer bir ifadeyle, dolaylı yöntemde doğması olası bir "bilgi kaybı" riski yoktur [82]. Çevirilerin karmaşıklığı açısından, doğrudan çevirinin dolaylı çeviriden daha basit olacağını söylemek çok güçtür. Bu tamamen sunucu ve alıcı formatları ile değişim formatına bağlıdır. Örneğin eğer sunucu ve alıcı formatları birbirine oldukça yakın ve değişim formatları bunlardan farklı ise bu durumda doğrudan çeviri daha kolay olacaktır.

Doğrudan yöntem, gerek çeviri kalitesi ve gerekse standart değişim formatlarının geliştirilmesi ve standartlaşmasında yaşanan yukarıda anılan problemler bakımından, dolaylı yöntemle üstünlük sağlamaktadır. Bu nedenle, konumsal veri değişiminde son yıllarda doğrudan yöntemle doğru bir eğilim gözlenmektedir [14], [82], [90]. Dolaylı yöntem ise, çok sayıda sistemin birbirleriyle veri değişiminde bulunabilmesi için gerekli çevirici sayısı ve çeviricilerin yenilenmesi ya da yeni çeviricilerin eklenmesi bakımından daha üstündür. Bu durumda, eğer doğrudan çeviriciler basit ve hızlı bir şekilde geliştirilebilirlerse, doğrudan yöntem konumsal veri değişimi için daha uygun görünmektedir. Çeviricilerin "anında" geliştirilebilmesi, aynı zamanda interoperabilite gereksinimlerinden biridir.

Doğrudan çeviricilerin basit ve hızlı bir şekilde geliştirilmesi için, derleyici ve yorumlayıcı teknolojilerinden yararlanılabilir. Bir programlama dili derleyicisinin görevi, programlama dilindeki bir programı makine diline çevirmektir. Bu, konumsal veri değişiminde iki farklı format arasındaki çeviriyi gerçekleştiren çeviricinin işlevine eşdeğerdir; Çünkü her iki durumda da söz konusu olan, iki ayrı dil arasında bir çeviridir. Kaldı ki derleyicilerin otomatik olarak oluşturulması, Bilgisayar Bilimleri alanında en çok çalışılmış konulardan biridir. Bu durumda, konumsal veri değişiminde çeviricilerin otomatik olarak geliştirilmesi için derleyici teknolojilerinden yararlanılması, çok akılcı bir yol olarak ortaya çıkmaktadır [14].

Farklı formatlardaki metinsel dökümanlara yönelik çevirici geliştirmede derleyici teknolojilerinden yararlanılması amacıyla çeşitli çalışmalar bulunmaktadır [70]. Bunlardan biri, Ohio Eyalet Üniversitesinin yürüttüğü *Chamelon* projesidir [91]. Bu proje, iki formatın sözdizim tanımlarından çeviricilerin otomatik olarak geliştirilmesi için bir arabirim oluşturmayı hedeflemiştir. Chamelon projesi, kapsamı genişletilmiş olarak *Nagiya* projesi adı altında devam etmektedir. Konumsal veri değişiminde derleyici teknolojilerinden yararlanma çalışmaları, metinsel veri değişimindekine oranla çok daha azdır. Bunlardan en kayda değer olanı [70] ın çalışmasıdır.

Derleyici ve yorumlayıcılarda üç temel eleman bulunur. Bunlardan "tarayıcı" ve "ayrıştırıcı", her ikisinde de ortak olan elemanlardır. Derleyicilerdeki "kod üretici" nin yerine, yorumlayıcılarda "gerçekleştirici" bulunur (Şekil 20). Bir programlama dili derleyicisi ya da yorumlayıcısı, o dilin sözdizimine gereksinim duyar; Tarayıcı ve ayrıştırıcı, programın bu sözdizimine uyup uymadığını test ederler. Kod üretici, sözdizimine uygunluğu tarayıcı tarafından doğrulanmış programı makine diline çevirir. Yorumlayıcı kullanılması durumunda, sözdizimine uyan her satır anında gerçekleştirilir. Her iki yaklaşımın birbirine üstünlükleri bulunmaktadır [92].



Şekil 20. Derleyici ve yorumlayıcı yapısı.

Bir programlama dilindeki program ile onun makine (bilgisayar) dilindeki karşılığı arasındaki çeviri, konumsal verinin iki farklı gösterimi arasındaki çeviriye eşdeğerdir. Dolayısıyla, verinin her iki taraftaki gösterimleri ve transfer edildiği gösterim belirli bir sözdizime göre resmi olarak tanımlanabilir ve çeviriler bir şekilde yine resmi olarak tanımlanabilirse çevirici program modülleri—kodlayıcı, kod çözücü, ve çevirici—otomatik olarak üretilebilir [70], [93]. Üretilen kodun gerçekleştirimi ise, derleyici ya da yorumlayıcı modunda yapılabilir. Burada verinin transfer edildiği gösterim, bir "transfer formatı" na uyan bir gösterimdir. Transfer formatı bir değişim formatı, bir ara format, ya da karşı tarafın formatı olabilir.

Bir veri dosyasının sözdizimi, yukarıda belirtildiği gibi, BNF ve ASN.1 gibi RTD'ler kullanılarak tanımlanabilir. Diğer bir ifadeyle, sunucu ya da alıcının iç formatındaki bir dosyanın sözdizimi, RTD'ler yardımıyla tanımlanabilir. Asıl sorun, çevirilerin tanımlanması ve çeviri işleminin otomatik hale getirilmesidir. [14], çeviri işlemi için ilişkisel veri tabanı teknolojisinden yararlanmayı denemiştir. Bu yaklaşımın espirisi, çeviri işleminde ilişkisel veri tabanlarının destekledikleri SQL, QUEL vs. gibi veri işleme dillerinin olanaklarından yararlanmaktır. Ancak, söz konusu çalışmada otomatikleştirilebilen yalnızca kod çözücünün geliştirilmesi olmuş, çevirici ve kodlayıcının otomatik olarak geliştirilmesi mümkün olmamıştır. Bunun üzerine aynı araştırmacılar, çeviri için ilişkisel bir VTYS kullanmak yerine, "C" veri yapıları ile, bu yapılar üzerinde çeşitli ilişkisel işlemleri yerine getiren "C" fonksiyonları kullanmak yoluna gitmişlerdir. Bu yolla, aynı verinin "C" veri yapılarındaki ve bir metin dosyasındaki gösterimleri arasında dönüşümü sağlayan kodlayıcı ve kod çözümleyici otomatik olarak geliştirilebilmektedir. Bunun için yalnızca metin dosyasındaki gösterim, resmi olarak tanımlanmalıdır. Bununla birlikte, çevirici modülü için gerekli algoritmalar hala otomatik olarak geliştirilememektedir [70], [75].

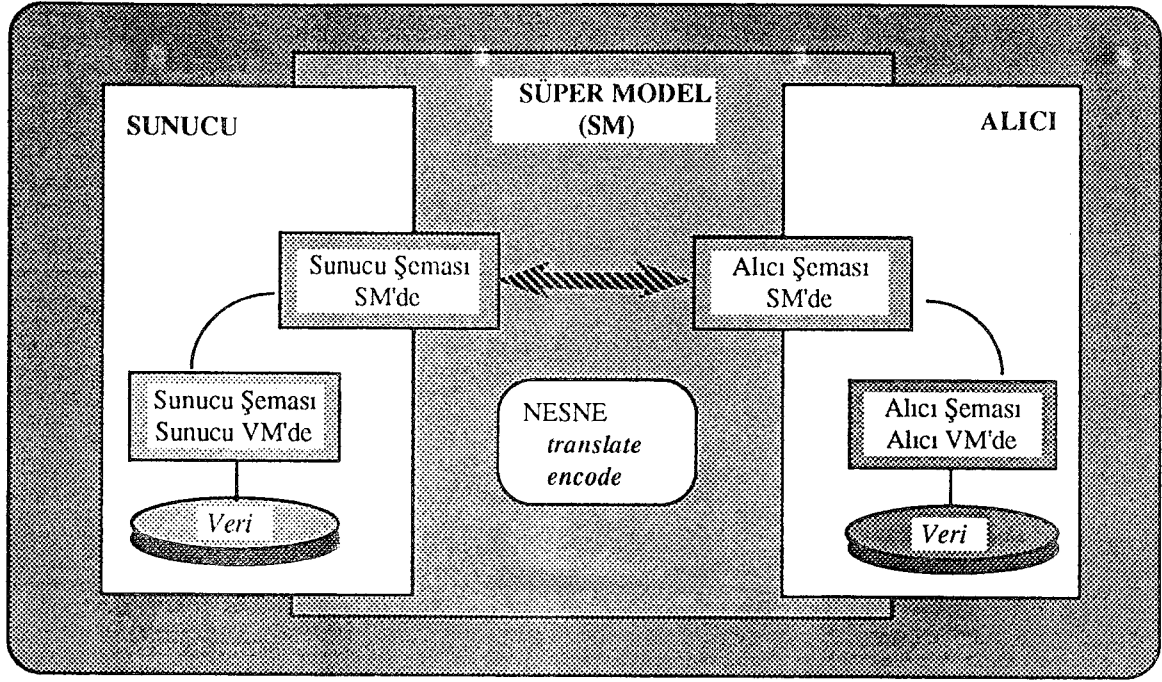
[81], konumsal veri değişimi için Nesneye Dayalı (ND) bir değişim ortamı önermiştir. Son yıllarda büyük bir popülerite kazanmış olan Nesneye Yönelim (NY) tekniğinin çıkış noktası, modüler ve genişletilebilir yazılım üretimidir. ND veri modelinde nesne tanımlamaları—Sınıflar— nesnelerin öznitelikleri yanında, nesnelere uygulanabilecek işlemleri—rutinler—de içerirler (bkz. kısım 1.2). Nesneye Yönelik Programlamada, programlar nesneler üzerinde tanımlı işlemlerin koşum zamanında ilgili nesnelere uygulanması ile yürütülür. Alışlagelmiş programlamadan oldukça önemli bir sapma ifade eden bu yaklaşım, konumsal veri değişimi açısından da çok önemli bir potansiyele sahiptir.

ND veri modeline dayalı bir değişim formatında, çeşitli veri tipleri arasındaki çeviriler veri tipi tanımlamaları içerisinde enkapsüle edilebilir. Dolayısıyla çeviri işlemi, çeviri rutinlerinin ilgili nesneler üzerine uygulanmasıyla gerçekleştirilebilir. Örnek olarak, sunucu

tarafında poligon sınırları çizgi parçaları ile, alıcı tarafında ise koordinat dizisi olarak gösterim edilebilir. Yani, alıcıda "çizgi-parçası" kavramı bulunmayabilir. Aradaki dönüşüm için gerekli çeviri rutinleri, tarafların poligon nesneleri içerisinde tanımlanabilir. Konumsal veri modellerinin farklılıklara karşın benzerlikler de taşıması nedeniyle, çeviri rutinlerinin önemli bir çoğunluğu farklı veri değişimlerinde ortak olarak gereksinim duyulacak rutinlerdir. Temel veri tipleri (INTEGER, REAL, STRING) arasındaki dönüşümler bunların en belirgin örneğidir. Benzer şekilde, farklı topolojik veri yapıları arasındaki çeviriler, birleştirme/ayırıştırma içeren çeviriler de farklı veri değişimlerinde kullanılabilir. Örneğin yukarıdaki poligon örneğindeki farklı poligon gösterimleri için gerekli çeviri rutini, diğer bakımlardan ne kadar farklı olurlarsa olsunlar, sözkonusu farklılığı gösteren bütün sistem ya da taraf çiftlerini içeren veri değişimlerinde kullanılabilir. Böylelikle, farklı veri değişimleri için ortak olan çeviri rutinleri tekrar tekrar kullanılabilen ve çeviriciler çok daha kolay ve hızlı bir şekilde geliştirilebilecektir [93]. NY yaklaşımın diğer avantajı, rutinlerin işlevlerinin nesneler bazında sınırlandırılması ile, alışlagelmiş programlama ortamlarında oldukça karmaşık bir iş olan rutin kitaplıkları organizasyonu yükünün ortadan kaldırılmış olmasıdır.

Böylelikle, eğer farklı veri tipleri arasında dönüşümü sağlayacak çeviri rutinleri, veri değişiminde bulunan tarafların alıp tekrar tekrar kullanabileceği ortak bir zeminde tanımlanabilirse, çeviriciler alışlagelmiş yöntemlere göre çok daha kolay ve hızlı bir şekilde geliştirilebilir. [81], bunun için bir "süper model" önermektedir. Süper model, aynen alışlagelmiş değişim formatları gibi çeşitli veri tipi tanımlarını, fakat bu formatlardan farklı olarak aynı zamanda, veri tipi tanımlamaları içerisinde çeviri rutin tanımlamalarını da içermektedir. Buna göre her nesne, diğer rutinler yanında bir "translate" ve "encode" rutini içermektedir (Şekil 21). Modelin "süper" olarak nitelenmesi, mümkün olduğunca çok sayıda farklı veri modeli arasında çeviriye olanak sağlayacak olmasındandır. Ancak böyle bir modelin tanımlanması kolay bir iş değildir.

ND veri değişim ortamında sunucu ve alıcı verisi, SM'in sunduğu kavramlar ve nesneler yardımıyla yeniden (SM cinsinden) tanımlanması söz konusudur. Bu tanımlamalar, Şekil 21 de SM'deki sunucu ve alıcı şemalarıdır. Çeviri ve kodlama işlemi bu şemalar üzerinden gerçekleşir. Çeviri ve kodlama işlemleri, nesnelere sırasıyla "translate" ve "encode" rutinlerinin uygulanmasıyla gerçekleşir. Şekil 21 den anlaşılacağı üzere, burada tarafların kendi şemalarındaki veri ile SM şemalarındaki veri arasında bir çeviri söz konusudur. O nedenle SM şemalarının oluşturulmasında, bu çevirinin mümkün olduğunca basit bir biçimde gerçekleştirilebilmesi dikkate alınır. Yukarıda değinildiği üzere, derleyici teknolojisinin ND veri değişim ortamında kullanılması ile, çeviriciler otomatik olarak geliştirilebilir [93]. Tezin ilgi alanı gereği burada daha fazla ayrıntıya girilmeyecektir. Ancak



Şekil 21. ND veri değişim ortamı.

şu kadarını belirtmek gerekir ki, burada tanımlandığı şekliyle bir süper modelin geliştirilmesi, konumsal veri yönetiminin geleceği açısından son derece önemli araştırma alanlarıdır.

1.4 Ulusal Konumsal Veri Altyapısı

Bilgi Teknolojisinin (BT) diğer alanlarında olduğu gibi, Konumsal Veri Yönetiminde de bilgisayar destekli sistemlerin "açık", "bütünleşik", "dağıtılmış", "federe" vs. olmalarının gerekliliği uzunca bir zamandır dile getirilmektedir. Böylesi sistemlere ulaşma yolunda önemli gelişmeler sağlanmış olmakla birlikte, hala çözülmesi gereken pek çok sorun bulunmaktadır. Söz konusu sistemlerin temellerini oluşturan ilkeler, "görevlerin dağıtılması", "kaynakların paylaşımı", ve fakat "ortaklaşa işlem" dir. Bütün bunların arkasındaki itici güç ise sürekli değişen gereksinimlerin hızlı, kolay, ve hesaplı bir şekilde karşılanabilmesidir. Burada "kaynakların paylaşımı" ile kastedilen donanım, yazılım, ve verinin paylaşımıdır. Tezin ilgi alanı gereği burada yalnızca veri paylaşımı üzerinde durulacaktır. Bu bölümde ayrıca veri paylaşımının ülke genelinde, etkin bir biçimde gerçekleştirilebilmesini sağlayacak olan Ulusal Konumsal Veri Altyapısı (UKVA) kavramı sunulmaktadır.

Konumsal Veri Yönetimi alanında ekonomik çözümler üretmenin yolunun, veri paylaşımından geçtiği öteden beri yaygın olarak kabul edilmekle birlikte, teknik ve kurumsal

problemler nedeniyle, veri paylaşımı alanında bugüne kadar sağlanan başarı hep sınırlı olmuştur [10], [18], [19], [94]. Gerek teknik ve gerekse kurumsal nedenlerle sınırlı sayıda kurum arasında bile yeterli koordinasyon ve işbirliğinin sağlanması çoğunlukla oldukça güç olmuştur [18], [95]. Bunda kurumsal sorunların payı çok daha büyük olmuştur. Konumsal veri ile çalışan kamu ve özel sektör kurumları arasında koordinasyonun sağlanarak, veri paylaşımında yaşanan teknik ve kurumsal problemlerin giderilmesi, ve sonuçta ülke düzeyinde bir "Bilgi Otobanı"nın oluşturulması için son yıllarda kavramsallaştırılan *konumsal veri altyapısı*nın A.B.D.'de oluşturulması gündemdedir. "Bilgi Toplumu" olma yolunda aşama kaydetmiş ülkelerin deneyimlerinden yararlanmak durumunda olan Türkiye'nin, süratle kendi Ulusal Konumsal Veri Altyapısını (UKVA) oluşturması kaçınılmazdır.

1.4.1 Veri Paylaşımı

Bir Coğrafi Bilgi Sistemi (CBS) kurma ve yaşatma girişiminde çoğu zaman göz ardı edilen bir gerçek, veri için yapılacak harcamaların, yazılım ve donanımın gerektirdiği yatırımdan çok daha fazla olduğudur. Her ne kadar sürekli gelişen teknoloji bugün, kapsam ve doğruluk olarak çok farklı özelliklerde konumsal veri eldesi için çeşitli araçlar sunmakta ise de, konumsal verinin ilk elden toplanması oldukça pahalıdır. Bazı tahminlere göre, bir CBS kurmada veri toplamının toplam maliyete oranı % 60-80 olarak kabul edilmektedir [8], [9]. Dahası, CBS yalnızca kurulma aşamasında değil, yaşamı boyunca sürekli konumsal veriye gereksinim duyar. Bu gereksinim bir uygulamanın gerektirdiği verinin veri tabanında bulunmaması, ya da istenen özellikleri sağlamaması (örn. güncel olmaması) nedeniyle olabilir. Bu açıdan [10], bir CBS'nin yaşam sürecinde donanım, yazılım, ve veri maliyetleri arasında yaklaşık olarak 1:10:100 oranının geçerli olduğunu belirtmektedir. Yüksek maliyet yanında, verinin ilk elden toplanması, bunun için gerekli süre ve eldeki projenin tamamlanma süresi dikkate alındığında kabul edilemez bir çözüm yolu olabilir.

Konumsal veriye Mühendislikten Pazarlamaya, Çevre Korumadan Arkeolojiye çok geniş bir yelpazede gereksinim duyulması nedeniyle, bir sistemin gereksinim duyduğu konumsal veri çoğu kez, ilk elden toplanması yerine sayısal formda başka kaynaklardan sağlanabilir. Örneğin hem tesis hatları yönetimi ve hem de ulaşım ağı planlamasında konumu tanımlayan veriye gereksinim vardır. Sayısal verinin farklı sistemler tarafından ortaklaşa kullanılabilmesi *veri paylaşımı* olarak anılmakta ve Konumsal Veri Yönetiminde ekonomik çözümler üretmenin yolu olarak kabul edilmektedir [10], [18], [19], [94]. Veri paylaşımı ile bugün çok yaygın bir problem olarak kabul edilen, farklı kurumların aynı veriyi ilk elden toplama yoluna giderek, kaynaklarını gereksizce israf etmeleri önlenecek ve veri toplama maliyeti önemli ölçüde azalacaktır. Ayrıca farklı kaynaklardan veri sağlanabilmesi ile yeni

uygulamalar mümkün olabilecek ve böylece çok daha çeşitli gereksinimlere cevap verebilecektir [94].

Genel olarak veri paylaşımının tarafları, konumsal veri ile iş yapan çeşitli bakanlıklara bağlı kamu kurumları, Belediyeler, özel sektör, ve Üniversiteler olabilir. Türkiye'de konumsal veri üreten ya da kullanan kamu kurumlarından bazıları Harita Genel Komutanlığı (HGK), İller Bankası, Karayolları, Tapu ve Kadastro, Devlet Su İşleri (DSİ), Devlet İstatistik Enstitüsü (DİE), Etibank, Köy Hizmetleri, Maden Tetkik Arama (MTA), Türkiye Petrolleri Anonim Ortaklığı (TPAO), Türkiye Elektrik Kurumu (TEK) genel müdürlükleri ve alt birimleridir. Gerek bu kurumlar, gerek özel sektör ve gerekse yerel yönetimler ilgi alanları gereği farklı özelliklerde konumsal veriye gereksinim duyarlar. Farklılık coğrafi, tematik, ve zamansal kapsam ile, doğruluk açısından olabilir; Örneğin TEK uygulamaları için yeterli olan doğruluk Tapu ve Kadastro uygulamaları için yeterli olmayabilir. Söz konusu farklılıklar bazen *veri uyumsuzluğu* olarak anılır. Veri uyumsuzluğu nedeniyle kurumlar bugüne kadar, gereksinim duydukları veriyi diğer bir kurumdan sağlama yerine, çoğunlukla ilk elden toplama yoluna gitmişlerdir.

Ancak, günümüz beklenti ve sorunlarının birbiriyle ilişkili ve bütünleşik doğası karşısında hızlı, sağlıklı, ve ekonomik çözümlere ancak, konumsal verinin gerek yatay ve gerekse dikey doğrultularda etkin bir biçimde paylaşılması ile ulaşılabilecektir. Örneğin orman planlarının hazırlanmasında vegetasyon haritası, toprak haritası, meşçere tipleri haritası, jeoloji haritası, topoğrafik haritalar, meteoroloji haritaları gibi farklı kurumlarca hazırlanan haritalara gereksinim vardır. Burada yatay doğrultuda veri paylaşımı ile kastedilen, farklı amaçlara hizmet eden ve farklı tipte veri ile çalışan kurumlar arasındaki veri paylaşımıdır. HGK'nın bir tarımsal ürün rekolte tahmini çalışması için DİE'ye, DİE'nin de Tarım Bakanlığına veri sağlaması gibi. Diğer bir örnek olarak mobil telefon baz istasyonlarının yerlerinin ve etki alanlarının belirlenmesi ile görevli bir kurum, hem topoğrafik hem de kadastral veri üreten kurumlardan veri transfer etmek durumunda kalacaktır. Dikey doğrultuda veri paylaşımı ise, verinin yerel, bölgesel, ulusal, ve hatta global düzeylerde paylaşımıdır. Bugün pek çok sorunun çözümü kurumların yetkisel sınırlarını aşmaktadır. Örneğin yerel idareler son yıllarda en önemli sorunlarından biri haline gelen katı atık yönetimine yönelik uygun yer seçimi için merkezi yönetimden veri istemek durumunda kalabilir. Tersine yönde merkezi yönetim, zehirli artıkların yönetimi ile ilgili ulusal politikalar üretmek amacıyla gerekli verileri yerel yönetimlerden sağlamak durumundadır.

1.4.2 Ulusal Konumsal Veri Altyapısı Gereksinimi

Konumsal veri ile çalışan kamu ve özel sektör kurumları arasında gerek yatay ve gerekse düşey doğrultularda etkin veri paylaşımının sağlanması ulaşılması öyle kolay bir hedef değildir. Bunun için, çeşitli kurum ya da kullanıcı gruplarının gereksinim duydukları veriye kolayca ulaşabilecekleri bir "Bilgi Otobanı"nın oluşturulması gerekmektedir. Bilgi otobanının oluşturulması ve yaşatılması, bir dizi teknik ve kurumsal problem içerir; Bilgi otobanının taraflara sağlayacağı yarar ve getireceği yük ne olacaktır? Tarafların fiziksel olarak birbirine bağlanmasında hangi teknoloji kullanılacaktır? Taraflar otoban için ne tür veri sağlayacak ve ne tür veriye hangi kısıtlamalarla erişim hakkına sahip olacaklardır? Bu ve benzeri sorunlar nedeniyle veri paylaşım programları bugüne kadar ancak sınırlı sayıda kurum arasında gerçekleştirilebilmiş, hatta az sayıda kurum arasında bile yeterli koordinasyon ve işbirliğinin sağlanması çoğunlukla çok güç olmuştur [8], [10], [17], [18], [19], [94], [95].

Genellikle, kurumsal problemlerin giderilmesi teknik problemlere göre çok daha zor olmuştur [10], [18], [19], [94]. Bunun en önemli nedenlerinden biri mevcut kurumsal yapılaşmanın günümüz sorunlarının gerektirdiği işbirliği için uygun olmamasıdır [18]. Bu bakımdan veri paylaşım programları yasal düzenlemeleri ya da belirli kurumların yeniden yapılanmasını gerektirebilir. Diğer bir neden, kurumların kendi içlerinde, yeni teknolojileri uyarlamakta oldukça yavaş olmalarıdır. Bu açıdan eksikliği duyulan, yeni teknolojilerin uyarlanmasına yönelik metodolojilerdir [10]. Dolayısıyla, kurumlar arası bir veri değişim programının gerek uygulamaya konması ve gerekse yaşatılmasında uygun teknoloji seçimi ve kurumsal problemlerin giderilmesi için metodolojilere gereksinim vardır.

Bir bilgi otobanının hizmet sunduğu alan ne kadar geniş olursa sağlayacağı yarar da o derece büyük olacaktır. Buna karşılık bilgi otobanının hizmet alanı genişledikçe, kurulması ve yaşatılması ile ilgili teknik ve kurumsal sorunlar daha karmaşık bir hale gelebilir. Ülke genelinde hizmet sunacak bir bilgi otobanı, konumsal veriye ulaşmada yaşanan kargaşa ve kaynak israfını gidererek, ülke ekonomisi için çok önemli bir katkı sağlayacaktır. Son yıllarda kavramsallaştırılan *Ulusal Konumsal Veri Altyapısı (UKVA)* ile, ülke genelinde etkin konumsal veri paylaşımını olanaklı kılacak bilgi otobanlarının oluşturulması hedeflenmektedir. UKVA ile konumsal veriye erişim ve kullanım önemli ölçüde iyileşecek ve ülke genelinde yatay ve düşey veri paylaşımı sağlanabilecektir. UKVA ile, Ulusal düzeyde bir veri paylaşım programının gerek hayata geçirilmesi ve gerekse sürdürülmesinde karşılaşılabilecek teknik ve kurumsal problemler daha kapsamlı ve tutarlı bir çerçevede ele alınarak, çözüm yöntem ve kuralları belirlenecektir [17], [18], [19]. Son bir kaç yıldır, başta Amerika olmak üzere pek çok ülke, bir UKVA kurma çabası içerisinde [19], [96]. Öyle ki

[19], UKVA'yı Amerika'nın Dünya pazarında ekonomik rekabet gücünü sürdürebilmesinde en önemli etken olarak görmektedir.

Türkiye penceresinden bakıldığında, konumsal veri yönetiminde CBS'lerin alışlagelmiş yöntemlerin yerini alması sürecinin yaşanmakta olduğu görülmektedir; CBS'ler Üniversiteler, kamu kurumları, yerel yönetimler, ve özel sektör için giderek daha da yoğunlaşan bir ilgi odağı durumundadır. Bir takım kamu kurumu ve belediyeler şimdiden CBS için gerekli yazılım ve donanım desteğine sahiptir. Diğer bir kısım, CBS'lere olan gereksinimin bilincinde somut adımlar atmaya düşünmektedir. Nihayet, özel sektör bu işin içindedir. Ancak, yeni bir teknolojinin en yararlı biçimde uyarlanabilmesi kuşkusuz, çeşitli sorunların çözülmesini içermektedir. Örneğin bir kurumun gereksinimleri doğrultusunda en uygun yazılım ve donanım nasıl belirlenecektir? Gerekli veri nasıl sağlanacaktır? Eğitilmiş personel gereksinimi nasıl karşılanacaktır? Kurumlar arası veri paylaşım stratejisi ne olmalıdır?

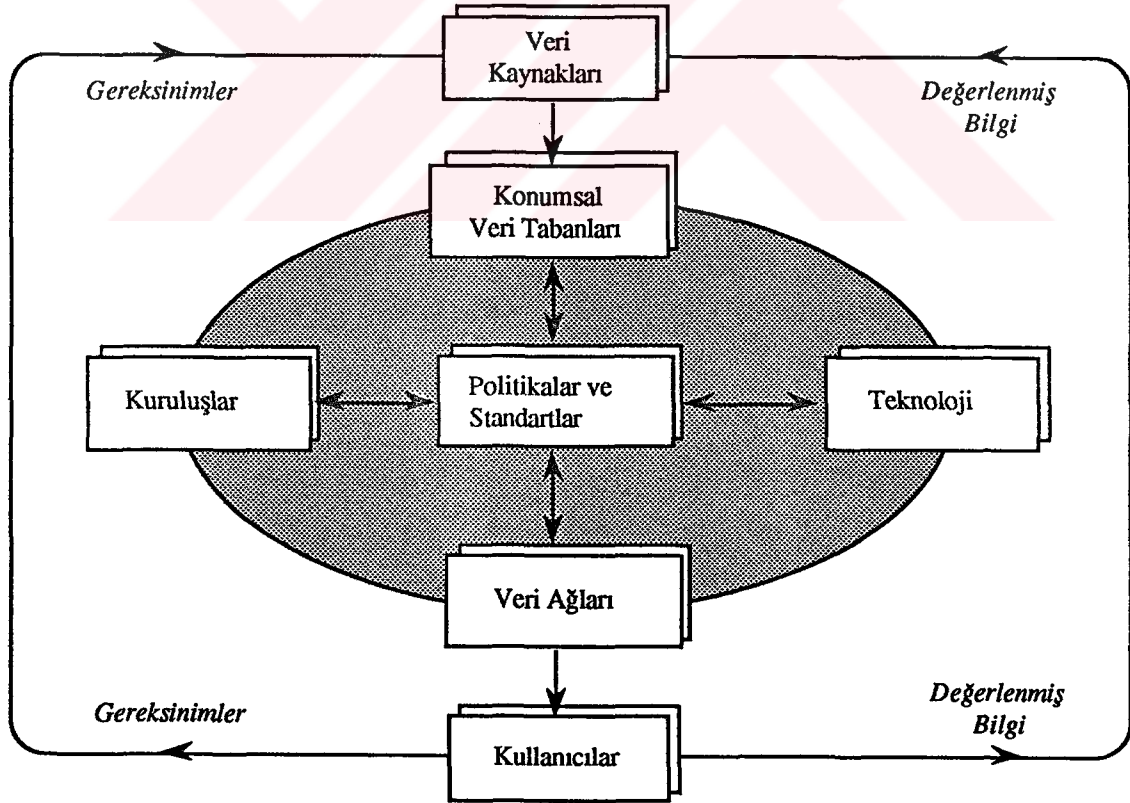
Türkiye, CBS'lere geçiş sürecinin henüz çok başlarında ve bu alandaki öncü ülkelere göre oldukça geri kalmış durumdadır. Örneğin CBS teknolojisinin ilk gerçekleştirmcileri olarak kabul edilen Kanada ve A.B.D., CBS çalışmalarına 1960'lı yıllarda başlamıştır [1], [97]. Bununla birlikte, bu ülkeler de bugün hala bütün sorunlarını çözebilmiş değildir. Veri paylaşımı, bu sorunların en önemlilerinden biri olmakla birlikte, yukarıda belirtildiği gibi bugün belirli bir noktaya gelinmiş olan bir konudur. CBS ve veri paylaşımının birbirinden ayıramaz kavramlar olmaları nedeniyle, CBS'lerin kullanıma girmeleri ile birlikte veri paylaşımı gündeme gelmiştir. Diğer bir anlatımla, CBS öncü ülkeleri veri paylaşımı alanında çok geniş bir deneyime sahiptir. Bu ülkelerde çeşitli veri paylaşım programlarının 1970 ve 1980 li yıllarda uygulamaya konulduğu ve bu programlarda belirli ölçülerde başarı sağlandığı görülmektedir [1], [19]. Türkiye'nin yapması gereken, CBS'ler alanındaki geri kalmışlığını, CBS teknolojisinde öncü, bu teknolojinin uyarlanması sürecinde pek çok sorununu çözmüş ülkelerin deneyimlerinden yararlanarak bir an önce gidermektir.

Veri paylaşımına olan gereksinim Türkiye koşullarında da aynen geçerlidir. Örneğin farklı kurumların aynı veriyi paylaşmak yerine ilk elden toplama yoluna gitmeleri, Türkiye'de de geleneksel bir rahatsızlıktır. Benzer şekilde, konumsal veri ile iş gören pek çok kamu kurumunun gerek kendi içlerinde ve gerekse diğer kurumlarla ilişkilerindeki "hantallık", öteden beri şikayet konusudur. Türkiye, veri paylaşımı alanında CBS teknolojisinde öncü ülkelerin bugün ulaştığı noktaya, onların geçtiği engebeli yolları aynen izleyerek ulaşmak zorunda değildir. Esasen Türkiye'nin buna tahammülü yoktur. O nedenle, Türkiye kendi UKVA'sını bir an önce kurmalıdır. UKVA ile aynı verinin farklı kurumlar tarafından ilk elden toplanması önlenmelidir. UKVA çerçevesinde ayrıca, konumsal veri ile

çalışan bir takım kamu kurumlarının "Bilgi Çağı" nın gerektirdiği dinamizmi yakalamaları yolunda çoktandır gereksinim duyulan "yeniden yapılanma"sı gerçekleştirilebilecektir. Çünkü UKVA, söz konusu kurumların işleyişi ve yapısı ile ilgili bir takım yasal düzenlemeleri gerektirecektir. Özetle UKVA, Türkiye'nin bir "Bilgi Toplumu" olabilmesi için kaçınılmazdır.

1.4.3 Ulusal Konumsal Veri Altyapısı

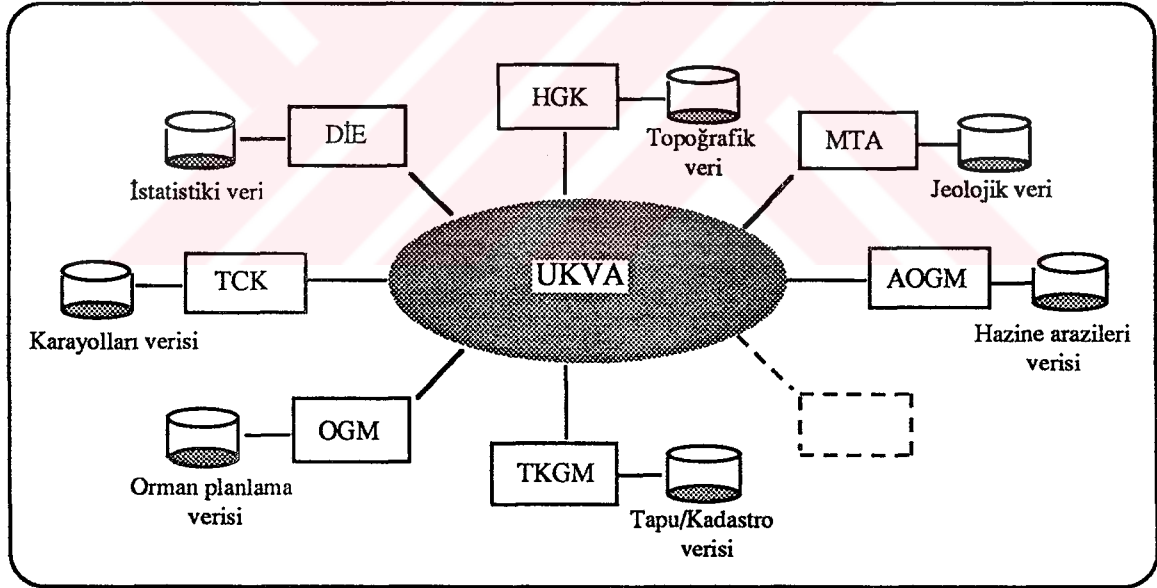
Ulusal Konumsal Veri Altyapısı (UKVA) bileşenleri veri kaynakları, konumsal veri tabanları, kurumlar, teknoloji, politikalar ve standartlar, veri ağları, ve kullanıcılardır (Şekil 22) [18], [98]. Şekil 22'de "Değerlenmiş Bilgi" ile, orijinal (kaynak) bilgiye yeni bilgilerin eklenmesiyle elde edilen bilgi kastedilmektedir. Örnek olarak bir kurumun başka bir kaynaktan temel haritalar sağlayarak, onlara turistik bilgiler eklemesi sonucunda elde edilen turistik haritalar verilebilir. UKVA ile bilgiye erişim olanakları çok önemli ölçüde iyileşeceğinden, değerlendirilmiş bilgi üretiminin önemli bir ticari sektör olarak ön plana çıkması beklenmektedir. UKVA bileşenleri, aşağıda özetle ele alınmaktadır.



Şekil 22. UKVA Elemanları [18] den.

1.4.3.1 Kurumlar

UKVA belirli bir grup, sektör, ya da coğrafi bölgeye özel olmayıp, ülke genelinde hizmet verecektir. Diğer bir anlatımla UKVA, yerel ve Ulusal düzeylerde çeşitli kamu kurumlarını ve özel sektörü içerecektir. Bu kamu kurumlarından bazıları Harita Genel Komutanlığı (HGK), İller Bankası, Karayolları, Tapu ve Kadastro Genel Müdürlüğü (TKGM), Devlet Su İşleri (DSİ), Devlet İstatistik Enstitüsü (DİE), Etibank, Köy Hizmetleri, Maden Tetkik Arama (MTA), Türkiye Petrolleri Anonim Ortaklığı (TPAO), Türkiye Elektrik Kurumu (TEK), Arsa Ofisi Genel Müdürlüğü (AOGM), Orman Genel Müdürlüğü (OGM) dır. Kurumlar, kendi ürettikleri veriyi diğer kurumların kullanımı için UKVA'ya sunacaklar, ve kendi gereksinimleri olan veriyi yine UKVA'dan sağlayacaklardır. Örneğin Tapu Kadastro Müdürlükleri tapu bilgilerini UKVA üzerinden kullanıma sunacak, belediyeler imar uygulamaları için gerekli bu bilgileri UKVA'dan sağlayarak kullanacaktır. Bu bakımdan UKVA'nın işlevi, ülke genelinde bir "Bilgi Pazarı" dır (Şekil 23). Bunun için her bir kurumun UKVA'ya ne sağlayacağı ve UKVA'dan ne alacağı, veri tipleri bazında belirlenmelidir.



Şekil 23. UKVA kavramı.

Veri paylaşım programlarının uygulanmasında kurumsal problemlerin aşılmasının, teknik problemlere göre daha zor olduğu yukarıda belirtilmişti. O nedenle UKVA'nın işlerlik kazanması, teknolojiden çok kurumsal düzenlemelere bağlıdır [19]. Bir kurumun sağlayacağı verinin, diğer kurumlar ve son kullanıcılar için ne kadar "açık" olacağı, kimin hangi tip veriyi sağlamak, güncelleştirmek, ve tutarlılığını temin etmekle yükümlü olacağı, UKVA'dan veri çekmenin bir kuruma maliyetinin ne olacağı, kurumsal düzenlemeler çerçevesinde çözülmesi gereken başlıca sorunlardan bir kaçıdır. Bu bakımdan UKVA,

kurumların görev ve sorumluluklarının yeniden belirlenmesini, dolayısıyla yeniden yapılanmalarını gerektirebilir. UKVA'nın kurulması ve yaşatılmasında gerekli koordinasyonun sağlanması için ayrıca, UKVA kapsamındaki kurumlardan birinin liderliği ya da bu kurumlar üzerinde yeni bir yapılanma gereklidir. Tezin ilgi alanı gereği burada bu yapılanmanın detaylarına girilmeyecektir. Ancak bu yapılanma ile ilgili bazı temel noktalara 1.4.4.4'de değinilmektedir.

1.4.3.2 Veri Kaynakları

Sürekli gelişen teknoloji bugün, konumsal veri eldesi için çok çeşitli araçlar sunmaktadır. Sayısallaştırıcılar, Uzaktan Algılama Sistemleri, Global Konum belirleme Sistemleri (GPS), Sayısal Fotogrametrik Sistemler, Sayısal Görüntü İşleme Sistemleri içerik ve doğruluk olarak oldukça farklı özelliklerde konumsal veri eldesi için çok çeşitli seçenekler sunmaktadır. Kullanıcı gereksinimleri veri kaynaklarını belirler. Örneğin bir orman yangınının ya da ormanı tehdit eden bir hastalığın izlenmesinde hava ya da uydu algılama sistemi görüntülerinin kullanılması ile daha sağlıklı sonuçlar elde edilebilir. Buna karşın, projenin finansal kısıtlamaları buna elveremeyebilir. Çünkü bugün bir LANDSAT TM görüntüsü 5000 Dolar civarındadır. Öte yandan, aynı uygulama için bölgenin topoğrafik yapısı ve kadastral durumu gerektiğinde mevcut çizgi haritaların sayısallaştırılması eğer haritalar güncelse uygun olabilir. Aksi takdirde yersel ya da fotogrametrik yöntemlere başvurulabilir. Çizgi haritaların sayısallaştırılması, konumsal veri toplamada en sık başvurulan yöntem olmakla birlikte, haritaların doğru ve güncel olmalarını gerektirir. Bu açıdan sayısallaştırma Türkiye koşullarında çoğu proje için uygun yol olmayabilir. Örneğin grafik yöntemle hazırlanmış kadastro paftalarının sayısallaştırılması, uygun yöntemler kullanılmadığı sürece anlamsız olacaktır.

1.4.3.3 Teknoloji

Teknoloji, başlıca dört açıdan ele alınabilir; iletişim teknolojisi, bilgisayar donanımı, yazılım, ve standartlar [18]. Veri paylaşımında geçmişte önemli teknik engellerden biri olan düşük veri transfer hızı, iletişim teknolojisindeki gelişmelerle önemli ölçüde problem olmaktan çıkmıştır. Bugün B-ISDN tipi iletişim ağları 155 Mbps (Mega bit/saniye) lik bir transfer hızı sunmakta, fiber-optik teknolojisi ile 13.22 Gbps (Giga bit/saniye) hedeflenmektedir [99]. Bunlar halen en yaygın bağlantı türü olan, analog telefon hatları ile ulaşılabilen 19.2 Kbps lik hızın çok üzerinde değerlerdir. Kaldı ki, 13.22 Gbps değeri iletişim teknolojisinin nihai hedefi değildir.

Benzer şekilde, bilgisayar donanımı alanında da baş döndürücü gelişmeler yaşanmakta, performans her geçen gün hızla artarken maliyetler düşmektedir. Öte yandan

donanım alanındaki gelişmelerin hızına ayak uyduramamakla birlikte, yazılım sektöründe de yeni eğilimler ortaya çıkmaktadır. Konumsal veri yönetimi açısından en çok yankı uyandıran eğilimlerden biri, özellikle geniş çaplı yazılımlar için, alışlagelmiş programlamaya göre çok daha yüksek derecede genişletilebilirlik vadeden Nesneye-Yönelim tekniğinin son on yıldaki çıkışıdır. Nesneye-Yönelik Programlama ve Nesneye-Dayalı Veri Tabanları henüz aşılamamış bir takım sorunlarına rağmen konumsal veri yönetimi alanında önemli katkılar sağlayacaktır [100]. Donanım ve yazılım alanındaki gelişmelerin desteğiyle, *Yapay Zeka* ve *Dağıtılmış Sistemler* gibi yeni bir takım ufuklar açılmıştır.

Standartlar, farklı donanım ve yazılımlara sahip sistemlerin birbiriyle "konuşabilmesi" için gereklidir. Bu açıdan Uluslararası Standartlar Örgütü (ISO)'nun OSI (Open Systems Interconnection) modeli çok önemli bir katkı sağlamıştır. ISO/OSI modeli, iki sistem arasındaki iletişim için yedi ayrı düzeyde standartlaşmayı önermektedir. Bu düzeyler düşük düzeyden yüksek düzeye doğru sırasıyla, fiziksel, veri linki, ağ, tur, sunuş, ve uygulama düzeyleridir [69]. TCP (Transfer Control Protocol)/IP (Internet Protocol) 3. ve 4. düzeyler için, Ethernet ise 1. ve 2. düzeyler için geçerli olan standartlardır. Bu konuda daha ayrıntılı bilgi Bölüm 2 de verilmektedir.

UKVA'nın teknolojiden beklentisi, kendi başlarına "akıllı" ve birbiriyle "konuşabilen" sistemlerdir. Bu beklentilerin ne dereceye kadar karşılanabileceği genel olarak Yapay Zeka ve Dağıtılmış Sistemler alanlarındaki gelişmelere bağlı olacaktır. Her iki alanda da önemli gelişmeler sağlanmış olmakla birlikte, henüz istenen düzeye ulaşamamıştır. Örneğin Dağıtılmış Sistemler alanında, konumsal veriye yönelik, yazılım ve donanım olarak farklı, kendi içlerinde özerk sistemlerin "bütünleşik" bir yapıda çalışabilmeleri özellikle veri organizasyonundaki problemler ve gerekli arabirimlerin eksikliği nedeniyle sorunludur [15], [101], [102]. Bununla birlikte, mevcut teknoloji gerek yazılım, gerek donanım, ve gerekse iletişim açısından çok çeşitli seçenekler sunmaktadır. Önemli olan, kurumların gereksinimlerine en uygun teknolojiyi seçmeleri ve onu en verimli bir biçimde kullanmalarındır.

1.5.3.4 Konumsal Veri Tabanları

UKVA yerel ve ulusal düzeylerde farklı ilgi alanlarına sahip çok sayıda kamu ve özel sektör kuruluşunun veri gereksinimini karşılamak durumunda olduğundan, farklı tipte veriler içeren çeşitli veri tabanlarından oluşacaktır (Şekil 23). Kuruluşların mevcut ya da UKVA kapsamında yeniden belirlenecek faaliyetleri, hangi veri tabanlarını oluşturmak ve güncelleştirmekle yükümlü olacaklarını belirleyecektir. Buna göre örneğin HGK, 1/25 000, 1/10 000, ve 1/5 000 ölçeğinde topoğrafik harita verisi, TKGM tapu kadastro verisi

sağlayabilir. UKVA'nın bir an önce hayata geçirilebilmesi için, kurumların gereksinimleri doğrultusunda önceliklerin belirlenerek, veri tabanlarının bu öncelik sırasında oluşturulması gerekir. Veri tabanlarının oluşturulmasında ayrıca, kurumların kapsam ve doğruluk olarak farklı özellikte veriye gereksinim duymaları sonucu ortaya çıkan veri uyumsuzluğu problemi, uygun düzenlemelerle giderilmelidir. Bunun için, veri sağlamakla görevli bir kurum, diğer kurumların gereksinimlerinin karşılanabilmesi için kendi gereksiniminden daha fazla veri üretmek durumunda kalabilir.

İdeal olarak, bir UKVA kullanıcısı bir Grafik Kullanıcı Arabirimi (GKA) ya da diğer destekleyici yazılımlar (araçlar) yardımıyla, gereksinim duyduğu verinin içerik ve karakteristiklerini belirterek, ilgili veri tabanına ulaşır, istediği bilgiyi edinir ya da gerekirse veriyi ayrıca işlemek üzere transfer eder. Örnek olarak, Ankara'da bir daire satın almak isteyen Trabzon'lu bir alıcının, daire hakkında ayrıntılı bilgi almak isteğiyle bir emlak alım-satım bürosuna başvurusu ele alınsın. Trabzon'daki büro yetkilisi, bu isteği belirli bir formda (örn. *coğrafi bölge + istenen bilgi türü*) UKVA'ya yönelterek önce, Ankara'daki ilgili kadastro müdürlüğü veri tabanına, oradan da dairenin bulunduğu parseli içeren paftaya erişerek kadastral durum ve tapu bilgilerini grafik ekranında alıcıya sunar. Bu senaryonun gerçekleşebilmesi için gerekli altyapı aşağıdaki gibi özetlenebilir:

Öncelikle, kullanıcı gereksinim duyduğu verinin içerik ve karakteristiklerini standart bir formda (örn. SQL kullanarak) belirtmelidir. Bunun için, UKVA verisinin de içerik ve karakteristikler olarak standart bir formda tanımlanmış olması gerekir. Bu, ilgili veriyi sağlayan kurumun sorumluluğuna bırakılabilir. Verinin "içerik ve karakteristikler"i meta veri olarak anılır ve verinin ait olduğu coğrafi bölgeyi, doğruluğunu, hangi kaynaktan, nasıl üretildiğini vs. belirtir. Meta verinin standart bir şekilde belirtilebilmesi için meta veri standartlarına gereksinim vardır. Bölüm 2'de görüleceği gibi meta verinin standart olarak belirtilmesi konumsal veri değişim standartları kapsamında düşünülebilir. İkinci gereksinim, aranan veri için hangi veri tabanlarına ya da veri kaynaklarına başvurulabileceğini kullanıcıya ileten "bilgi sunucuları" dır. Ayrıca gerek bilgi sunucuları ve gerekse bu sunucular üzerinden ulaşılacak veri kaynakları kataloglanmalıdır. Üçüncü olarak, verinin transferinde kaynak formatından kullanıcı formatına dönüşüm için çeviricilere gereksinim vardır. Söz konusu çeviri işlemi kaynak ve kullanıcı formatları arasında doğrudan ya da bir değişim formatı üzerinden gerçekleştirilebilir. İkinci durumda, UKVA için geçerli konumsal veri değişim formatı sayısı sınırlı tutulmalıdır. Nihayet, bir kullanıcının UKVA'ya çıkışını, dolaşımını, ve ilgili veri kaynaklarına erişimini sağlayacak ve düzenleyecek yazılım ve protokoller gereklidir. Bunun için WAIS, Gopher, Mosaic, Netscape gibi internet'de söz konusu işlevleri sunan ürünler, UKVA için uyarlanmalı ya da tamamiyle konumsal veriye yönelik yenileri geliştirilmelidir.

1.4.3.5 Veri Ağları

Veri ağları ile, UKVA kullanıcılarını ve veri tabanlarını birbirine bağlayan otoyollar kastedilmektedir. UKVA, çok sayıda Yerel Alan Ağı (LAN) ve Geniş Alan Ağı (WAN)'ndan oluşacaktır. LAN'lar bir kaç km'lik bir alana yöneliktir. Üniversite yerleşkelerini kapsayan ağlar, bir kurum binası içerisindeki ağlar tipik örneklerdir. Buna karşılık WAN'lar ile amaçlanan kent, ülke, ve Dünya çapında (Internet ve Usenet gibi) iletişim ağlarıdır. İletişim Teknolojisi'nin iletişim hızı, güvenlik, güvenilirlik, ve maliyet açısından WAN için sunduğu seçenekler, LAN'dakinden çok daha fazladır; Örneğin iletişim hızı açısından seçenekler LAN için, 10 Mbps ve 100 Mbps ile sınırlı iken, WAN'da seçenek sayısı çok daha fazladır. Bu durum, WAN'ların LAN'lara göre çok daha geniş alanlara yönelik olmaları nedeniyle, LAN teknolojisinin WAN için uygun olmaması ve WAN için bugüne kadar çeşitli arayışlara girilmiş olmasının sonucudur. O nedenle, WAN için uygun teknoloji seçimi, LAN'dakinden çok daha kritik kararlar gerektirir. İletişim Teknolojisi'nin WAN'a yönelik seçenekleri Tablo 1 de özetlenmiştir. Tablo 1 de yer alan ATM, SMDS, ve

Tablo 1. Geniş Alan Ağları (WAN) için seçenekler.

Teknoloji	Tanım	İletişim Hızı
Analog telefon hatları	Analog telefon hatları, hat uçlarında analog-dijital dönüşümü için modemler	Teorik 30 Kbps, Pratik 9.6 - 19.2 Kbps
Özel, kiralık hatlar	Telefon hatları üzerinde noktadan-noktaya özel olarak ayrılmış hatlar, hat uçlarında sayısal servis birimleri (DSU) ile sayısal iletim	2.4, 4.8, 9.6, 19.2, 56, ve 64 Kbps
Özel, T1 ve T3	Noktadan-noktaya, özel, sayısal, çok kanallı (duruma göre birleştirilip ayrılabilen) hatlar	T1 15 Mbps, T3 yaklaşık 45 Mbps
ISDN (Integrated Systems Digital Network)	Bütünleşik ses ve veri iletişimine yönelik, çok kanallı, sayısal ağlar	64 Kbps
B-ISDN (Broadband ISDN)	ATM ve SONET teknolojilerine dayalı iletişim	155 Mbps
SMDS (Switched Multimegabit Data Service)	T1 ve T3 üzerinden, her bir veri hücresi adresini içerir	56 Kbps - 45 Mbps
Frame Relay	Yalnızca gönderici ve alıcı uçlarda hata kontrolü	64 Kbps - 2 Mbps (ve üstü)
ATM (Asynchronous Transfer Mode)	Fiber-optik hatlar üzerinden, "veri hücreleri" ile sayısal veri, video, ve ses iletimi	25 Mbps - 1.2 Gbps (ve üstü)
SONET (Synchronous Optical Network)	Fiber-optik hat üzerinden, değişen iletimli veri dizileri birleştirilebilir veya ayrıştırılabilir	51.84 Mbps - 13.22 Gbps
Kablosuz iletişim	Mikro dalga ve Radyo dalgalarına, uydu, alıcı, ve gönderici sistemlerine dayalı iletişim	Mikrodalga için 64 Kbps'lik çoklu kanallar, Radyo dalgaları için 9.6 Kbps

SONET teknolojileri henüz tamamiyle işlevsel değildir. Bununla birlikte, A.B.D. ve Kanada'da özellikle ATM ve SMDS teknolojilerinin 1-2 yıl içerisinde bazı eksikleri ile işlevsel hale gelmesi beklenmektedir. SONET ise henüz geliştirilme aşamasında olup, ancak sınırlı düzeyde uygulanabilmektedir [99].

İletişim teknolojisinin geniş alan iletişimi için çok çeşitli seçenekler sunması karşısında önemli olan, UKVA taraflarının iletişim hızı, güvenlik, güvenilirlik, yaygınlık açısından gereksinimleri ile, maliyetlerin dikkate alınarak en uygun ağ teknolojilerinin belirlenmesidir. İletişim hızı, maliyet ve yaygınlık birbiriyle çok yakından ilişkilidir. Genellikle yüksek iletişim hızı, yüksek maliyet anlamına gelir. Bir ağın maliyeti ise doğrudan ağın yaygınlığını, hizmet alanının belirlenmesinde çok önemli bir rol oynayacaktır. Bir veri tabanına, farklı kesimlerden çok sayıda kullanıcının giriş yapması sonucunda, o veri tabanının güvenliğinin tehlikeye gireceği açıktır. Önlem olarak, yalnızca yetkili kullanıcıların veri tabanına girebilmesi ve yine ancak belirli kullanıcıların veri tabanında güncelleştirme yapabilmesi gerekir. Bunun için, verinin özel olarak kodlanması (data encryption) yoluna gidilebilir [103], [104]. Güvenilirlikten kasıt, ağda iletişimin sağlıklı bir biçimde gerçekleşmesidir.

İletişim hızı açısından bakıldığında ise, konumsal veri ile çalışan kurumların artık çok farklı kaynaklardan veri sağlamak durumunda olması ve bu verinin çok önemli bir kısmının raster veri olarak gelmesi sonucu bugün herkesin hız istemekte olduğu görülmektedir [104]. 6 MB'lık bir LANDSAT görüntüsünün 9.6 Kbps lik bir telefon hattı aracılığıyla transferinin yaklaşık bir buçuk saat alması, çoğu CBS kullanıcısı için kabul edilebilir değildir. [99]'un A.B.D.'de 86 kuruluş üzerinde yaptığı araştırmaya göre, kuruluşların çoğu kullandıkları iletişim teknolojilerini, iletişim hızı çok daha yüksek yeni teknolojilerle (başta ATM gibi) yenilemeyi planlamaktadır. Bununla birlikte, iletişim hızı gereksinimi, kurumdan kuruma, uygulamadan uygulamaya farklılık gösterecektir [104]. Örneğin hava durumu tahmini, askeri amaçlı uygulamalar ve acil durum yönetimi, sıradan CBS uygulamalarına göre daha yüksek transfer hızı gerektiren uygulamalardır.

1.4.3.6 Politikalar ve Standartlar

UKVA'nın kalbi durumundaki politikalar ve standartlar diğer bütün bileşenleri etkilerler [19]. Politikalar, UKVA'nın geliştirileceği ve yönetileceği ortamı hazırlar; hedefleri ve bu hedeflere nasıl ulaşılabileceğini belirler. Kurumların UKVA içerisindeki görev ve sorumluluklarının belirlenerek gerekli finansal desteklerin sağlanması, bilginin açıklığı, maliyeti, ve sahipliği politikalar bağlamında ele alınması gereken konulardan bazılarıdır. UKVA bunların çok geniş bir çerçevede ele alınmasını gerektirmektedir. Bilgi kime ne kadar açık olacaktır? Yani isteyen istediği bilgiye ulaşabilecek midir? Bir vatandaş doğrudan

UKVA veri tabanlarına ulaşabilecek midir, yoksa bunun için görevli personelin aracılığı mı gereklidir? Bilginin doğruluğunu ve güncelliğini kim sağlayacaktır? Standartları kim belirleyecek ve bunlara uyulmasını kim, nasıl sağlayacaktır? Bilgi ücretsiz mi olacaktır, değilse fiyatlar nasıl belirlenecektir? UKVA ile elde edilen gelir nasıl dağıtılacaktır? Bunlar, politikaların belirlenmesinde çözümlenmesi gereken soruların başlıcalarıdır.

UKVA kapsamında standartlar, bu tezin ilgi odağını oluşturan konumsal veri değişimi ile sınırlı olmayıp, donanım, yazılım, iletişim, ve veri tanımlamayı içerir [17]. UKVA'nın esaslı farklı yazılım, donanım, veri gösterimi, ve iletişim sistemlerini kullanan taraflar arasında "iletişim" olduğu için, standartlar hayati önem taşımaktadır. Donanım, yazılım, ve iletişim'de standartlaşma yönünde bugün belirli bir noktaya gelinmiş olmasına karşın, konumsal veri değişiminde standartlaşma, gerekliliği öteden beri kabul edilmekle birlikte bir türlü sağlanamamış ve Konumsal Veri Yönetiminde en çok baş ağrıtan sorunlardan biri olmuştur [98]. Gerek teknik ve gerekse teknik olmayan nedenlerle, veri değişimi için herkesin kendi standardını tercih etmesi sonucu, konumsal veri değişiminde bugün çok çeşitli "standartlar" kullanılmaktadır. Bunun yarattığı en önemli sorun, bir sistemin farklı kaynaklardan veri transfer edebilmesi için çok sayıda formatı tanınması gerekmektedir. Bu da, 1.3 te görüldüğü gibi sorun yaratmaktadır. O nedenle UKVA'da kullanılacak konumsal veri değişim formatı sayısı bir ya da birkaç ile sınırlı tutulmalıdır. Konumsal veri değişiminde kullanılan değişim formatlarının sayıca sınırlandırılması yönünde, son yıllarda ulusal ve uluslararası düzeyde yaşanan gelişmeler (bkz. kısım 1.5) umut vericidir. Konumsal veri değişiminde standartlaşma konusu 1.3 te daha ayrıntılı olarak ele alınmaktadır.

1.4.3.7 Kullanıcılar

Genel olarak kullanıcılar, teknik ve son kullanıcılar olmak üzere iki grupta düşünülebilir. Teknik kullanıcılar, belirli düzeyde yazılım bilgisine sahip, çalıştıkları kurumda ilgili yazılımları kullanabilen ve uygulamaları için başka kaynaklardan veriye gereksinim duyan kullanıcılardır. Son kullanıcılar ise, yüksek düzeyli bir arabirim yardımıyla sistemle iletişim kuran, her hangi bir konuda bilgi edinmek isteyen, sıradan kullanıcılardır (örn. parseli ile ilgili tapu ve kadastro bilgilerini öğrenmek için bir kadastro müdürlüğüne baş vuran vatandaş). Bu tür kullanıcıların yalnızca, sistemle iletişimlerini sağlayan arabirimi kullanabilecek kadar yazılım bilgisine sahip olması beklenir. O nedenle, son kullanıcılar için yüksek düzeyli arabirimlere (mümkün olduğunca doğal bir iletişime olanak tanıyan) gereksinim vardır. Örneğin parseline ait tapu/kadastro bilgilerini öğrenmek isteyen vatandaş için, parselin numarasını ve bulunduğu bölgeyi bilmesi yeterli olmalıdır. UKVA'nın kullanıcılar için sunması gereken, UKVA'sız yapmakta olduklarını daha kolay ve ekonomik bir şekilde yapabilmeleridir. Aksi takdirde UKVA'nın gerekçesi kalmayacaktır

[19]. Bu nedenle kullanıcılar, UKVA'nın oluşturulması ve yaşatılması sürecinde göz ardı edilmemesi gereken çok önemli bir unsurdur.

1.4.4 Ulusal Konumsal Veri Altyapısı'nın Oluşturulması

Tezin ilgi alanı gereği bu kısımda amaç, UKVA'nın oluşturulması ve yaşatılması için yapılması gerekenleri bütün ayrıntılarıyla, adım adım sıralamak değildir. Yalnızca UKVA'nın oluşturulması ve yaşatılması ile ilgili konular genel bir çerçevede ele alınacaktır. Bunların bir kısmına yukarıda, UKVA bileşenleri açıklanırken çok kısa olarak dokunulmuştur.

UKVA'nın kurulabilmesi için her şeyden önce UKVA taraflarının desteklerin kazanılması gerekir. Diğer bir anlatımla kamu kurumları, yerel yönetimler, özel sektör, ve üniversitelerin UKVA'yı anlamaları ve benimsemeleri sağlanmalıdır. Kuşkusuz tarafların UKVA'yı benimsemeleri, O'nu teşvik edici bulmaları ile olanaklı olacaktır. Veri paylaşımı ile konumsal veri elde etme maliyeti çok önemli boyutta azalacaktır. Buna karşılık, UKVA ile kurumların yükümlülükleri artabilecektir. Örneğin UKVA'ya veri sağlamakla görevli "X" kuruluşu, "Y" kuruluşunun istediği özellikte verinin sağlanması için, kendi gereksiniminden daha fazla veri toplamak durumunda kalabilir. Dahası bu, "Y" kuruluşu açısından da diğer kuruluşa bağımlılık anlamına geleceği için, kabul edilemez bir durum olabilir. O nedenle, UKVA'nın taraflar için sağlayacağı yararlar ve getireceği yükümlülükler açık bir şekilde ortaya konmalıdır.

UKVA'nın kurulması ve yaşatılması için bir dizi gereksinimin karşılanması gerekir. Söz konusu gereksinimler politik, yönetsel, ve teknik olmak üzere üç ana grup altında düşünülebilir [18]. Politik gereksinimler, UKVA'nın kurulması ve yaşatılması ile ilgili temel politikaların belirlenmesine yöneliktir. Yönetsel gereksinimler, UKVA'nın gerek hayata geçirilmesi ve gerekse yaşatılma sürecindeki yönetimi ile ilgilidir. Teknik gereksinimler ise, UKVA'nın gerektirdiği teknik altyapının oluşturulması ve bakımı içindir. UKVA gereksinimleri aşağıda özetle ele alınmaktadır.

1.4.4.1 Politik Gereksinimler

Politik gereksinimler çok geniş kapsamlı ve kısmen UKVA kapsamını aşan gereksinimlerdir. Genel olarak, UKVA kapsamındaki kamu kurumları, özel sektör, yerel yönetimler ve Üniversitelerin işlevleri ile görev ve sorumluklarının genel hatlarıyla belirlenmesi söz konusudur. Buna göre kamu kurumlarının kısmen de olsa yeniden yapılanmaları, dolayısıyla bir takım yasal düzenlemeler gündeme gelebilir. UKVA

taraflarının UKVA içerisindeki işlevlerinin belirlenmesinin pek çok boyutu vardır. Bunlardan en önemlileri şunlardır:

- UKVA'ya hangi kuruluşların hangi türde veri sağlayacağını belirlenmesi,
- Kurumların UKVA'nın geliştirilmesi kapsamında finansal kaynak gereksinimlerine yönelik ekonomi politikalarının belirlenmesi,
- Bilgi fiyatlandırma politikalarının belirlenmesi (UKVA üzerinden bilgi edinmenin maliyetini kim nasıl belirleyecek, elde edilen gelir nasıl paylaşılacaktır? gibi sorular söz konusudur.),
- Veri toplama ve yönetimi ile ilgili standartların belirlenmesi ve bunlara uygunluğun sağlanmasından sorumlu otoritelerin belirlenmesi,
- Ticaret ve iletişim alanlarında önceliklerin ve ilgili politikaların belirlenmesi,
- Bilginin sahipliği, gizliliği, özelliği, telif hakları, değerlendirilmiş bilginin sahipliği vb. gibi sorunların çözülmesi.

1.4.4.2 Yönetimsel Gereksinimler

Yönetimsel gereksinimler, UKVA için gerekli kurumsal yapılanmanın oluşturulması, gerçekleştirim stratejilerinin belirlenmesi, gerçekleştirimde rollerin dağıtımı ve projenin yönetimi ile ilgilidir. UKVA için gerekli kurumsal yapılanma, UKVA'nın geliştirilmesi ve yaşatılmasında liderlik rolünü üstlenecek bir kuruluş ile bağlı çalışma grupları şeklinde olabilir. Lider kuruluş UKVA'da yer alacak kamu kurumlarından biri de olabilir. Söz konusu yapılanma, UKVA'nın geliştirilme sürecinde değişikliğe uğrayabilir. Örneğin gerekli kurumsal destek ve bağlılıkların kazanılmasından sonra bu görevi üstlenmiş olan çalışma grubuna gerek kalmayacaktır. Temel gereksinimler şunlardır:

- UKVA için gerekli kurumsal yapılanmanın oluşturulması,
- Gerçekleştirim stratejilerinin (öncelikler vs.) belirlenmesi,
- UKVA'nın geliştirilmesi ve yaşatılmasındaki rollerin, hak ve yükümlülüklerin (örn. veri tabanlarının bakımına yönelik) UKVA taraflarına dağıtılması,
- Özel sektörün UKVA içerisindeki konumuna yönelik stratejilerin geliştirilmesi
- UKVA'nın teşvik edilmesi ve pazarlanması,

1.4.4.3 Teknik Gereksinimler

Teknik gereksinimler, sürekli gelişen teknolojinin çok yakından izlenerek UKVA gereksinimlerine uygun teknolojilerin seçimi ya da geliştirilmesi, seçilen teknolojinin en

verimli bir biçimde kullanılması ile, gerektiğinde yeni teknolojilerin uyarlanmasına yöneliktir. Başlıca gereksinimler şunlardır:

- UKVA veri tabanlarının, veri gruplarının ve bilgi sunucularının tasarlanarak, öncelik sırasında oluşturulması,
- Konumsal veri değişimi, meta veri, detay sınıflaması ve kartografik gösterim standartlarının ve konumsal veri değişim stratejilerinin (doğrudan veya dolaylı) belirlenmesi ya da geliştirilmesi,
- Kullanıcıların UKVA veri tabanlarını, veri gruplarını görme, sorgulama, veri transferi yapabilmelerine yönelik kullanıcı hizmetlerinin (arabirimler, görme, tarama, ve sorgulama araçları vs.) geliştirilmesi,
- Geniş alan ağı seçeneklerinin değerlendirilerek, gereksinimlere en uygun ağı konfigürasyonlarının belirlenmesi,
- UKVA'nın gerek kurulması ve gerekse yaşatılması sürecinde sürekli olarak, gereksinimlerin ve sınırlayıcı faktörlerin değerlendirilerek, teknik bakım stratejilerinin geliştirilmesi.

1.4.4.4 Stratejik Plan

Türkiye Bilgi Çağını yakalamak ve bir Bilgi Toplumu olmak istiyorsa, UKVA'yı bir an önce hayata geçirmelidir. "UKVA" Türkiye için oldukça yeni bir kavramdır. O nedenle UKVA için ilk adım, UKVA kavramının siyasi otoriteye, UKVA taraflarına, ve kamuoyuna anlatılarak, ulusal bir politika olarak benimsenmesini sağlamaktır. Kuşkusuz, UKVA öncelikle siyasi otorite tarafından desteklenmelidir. Ardından, UKVA taraflarının politik ve finansal desteklerinin kazanılması gerekir. Tarafların destek ve bağlılıkları başlangıçta çok güçlü, UKVA'nın yaşatılabilmesi için de sürekli olmalıdır.

UKVA'nın hayata geçirilmesi ve yaşatılması, UKVA'ya özel bir kurumsal yapılanma gerektirir. Tarafların politik ve finansal desteklerinin kazanılması, gerekli koordinasyonun sağlanması, gereksinimlerin karşılanması ve karşılaşılan sorunların giderilmesi bu kurumsal yapılanmanın temel görevleri olacaktır. Diğer bir anlatımla, bu kurumsal yapı UKVA'yı üstlenecek ve yürütecektir. Yukarıda da değinildiği gibi söz konusu kurumsal yapı, bir kurum (UKVA'da yer alacak kamu kurumlarından biri olabilir) ve ona bağlı çalışma grupları şeklinde olabilir. Bu kurumun yeterli yaptırım gücü olmalıdır. Bu bakımdan kurumun bir bakanlığa bağlanması düşünülebilir. Ancak UKVA'nın siyasi otorite değişikliklerinden etkilenmemesi ve kesintiye uğramaması için gerekli önlemler alınmalıdır. Gerek söz konusu kurumsal yapılanma ve gerekse UKVA'nın diğer boyutları için, başta A.B.D., Avustralya, ve Kanada olmak üzere, diğer ülkelerdeki UKVA çalışmalarını gözden geçirmek yararlı

olacaktır. Örneğin [19], UKVA için yetkili mercinin Ticaret Bakanlığı olmasını önermektedir. Bunun en önemli nedeni, UKVA ile ekonominin ulusal ve uluslar arası düzeyde yeni boyutlar kazanacak olmasıdır (Örn. "değerlenmiş-bilgi" üretimi çok kolay bir hale gelecektir). Diğer neden ise, özel sektörün de UKVA'nın bir parçası olmasıdır.

UKVA'nın bir an önce hayata geçirilebilmesi için, çalışmalar eş zamanlı olarak yürütülmelidir. Bir taraftan siyasi otorite gerekli politikaları belirlerken, diğer yandan UKVA'yı üstlenen kurum, yönetsel ve teknik gereksinimleri ele almalıdır. Bununla birlikte, gereksinimlerin karşılanması gerektiren gereksinimler arası (örn. iletişim ağı konfigürasyonları, kurumların işlevlerine göre belirlenecektir) ve gerekse gereksinim içi öncelikler (örn. veri tabanlarının oluşturulmasında) dikkate alınmalıdır.

UKVA'nın en önemli teknik gereksinimlerinden biri, konumsal veri değişimi, meta veri, detay sınıflaması ve kartografik gösterim standartları ile, konumsal veri değişim stratejilerinin belirlenmesi ya da geliştirilmesidir. Bu gereksinim, bu tezin ilgi alanını oluşturmuştur. Tezde, UKVA gereksinimlerini karşılayacak ve taraflar arasında etkin veri paylaşımına olanak tanıyacak bir veri değişim standardı önerilmektedir. Böylece, burada öngörülen genel sırada olmamakla birlikte, UKVA için ilk adım atılmış ve çok önemli bir gereksinim karşılanmış olmaktadır.

2. YAPILAN ÇALIŞMALAR

Genel ve aynı zamanda esnek bir konumsal veri değişim formatının oldukça kapsamlı ve karmaşık olacağı bir gerçektir. Bu nedenle konumsal veri değişim formatlarının değerlendirilmesi, belirli ölçütlere göre yapılmalıdır. Öte yandan bu tezde amaç UKVA gereksinimlerini karşılayacak bir konumsal veri değişim standardının belirlenmesi olduğundan, gereksinimi duyulan, bir konumsal veri değişim formatını bütün yönleriyle değerlendirecek ölçütlerdi. Bu alandaki başlıca çalışmalar olan [12] ve [16] de önerilen değerlendirme ölçütleri bu açıdan yetersizdi. O nedenle bu tezde, bir dizi değerlendirme ölçütü belirlenmiştir. Bu ölçütler bir konumsal veri değişim formatını "veri modeli", "gerçekleştirim", ve "değişim ortamı" açısından değerlendiren ölçütlerdir. Bu bölümde söz konusu ölçütler tanıtılmaktadır. Bu bölümde sıkça geçeceği için, "konumsal veri değişim formatı" yerine, "DF" (Değişim Formatı) kısaltması kullanılacaktır.

Değerlendirme ölçütleri, pek çok geleneksel konumsal veri değişim formatı dökümanında görülemeyen bir açıklıkla, veri modeli ve gerçekleştirim kavramlarını birbirinden ayırmaktadır. Veri modeli ve gerçekleştirim boyutlarının birbirinden ayrı tutulması, değişim formatının gerçekleştirim ortamı (donanım vs.) bağımsızlığının sağlanması açısından gereklidir. Bir konumsal veri değişim formatı, soyut tanımı yanında gerçekleştirim detaylarını da kapsamlı ve bir değişim ortamı ile desteklenmelidir. Geleneksel konumsal veri değişim formatlarında olduğu gibi, formatın yalnızca soyut bir tanım içermesi, formatın bilgisayar ortamında gerçekleştirimine ait ayrıntıların ve buna yönelik destekleyici yazılımların formatın kapsamı dışında düşünülmesi, formatın bir "standard" olarak kabul görmesi yolundaki en önemli etkenlerden biridir. Genel olarak, değerlendirme ölçütlerinin aradığı, veri modeli açısından genellik ve genişletilebilirlik, gerçekleştirim açısından esneklik ve donanım bağımsızlığı, ve değişim ortamı açısından çeviricilerin kolay ve hızlı bir şekilde geliştirilmesine ve etkin bir veri değişimine olanak tanıyacak servis ve araçlar ya da bunları içeren bir Grafik Kullanıcı Arabirimi (GKA) dir.

2.1 Veri Modeli Ölçütleri

Veri modeli ile ilgili olarak onbir ayrı ölçüt belirlenmiştir. Bunlar veri modeli tipi, raster veri, kartoğrafik temsil, koordinatlar, topoloji-geometri ayırımı, karmaşık nesne, nesne paylaşımı, meta veri, nesne bazında veri değişimi, zamansal veri, ve genişletilebilirlik

ölçütleridir. Veri modeli ölçütleri, bir CBS nin veri modelinin değerlendirilmesinde de kullanılabilir.

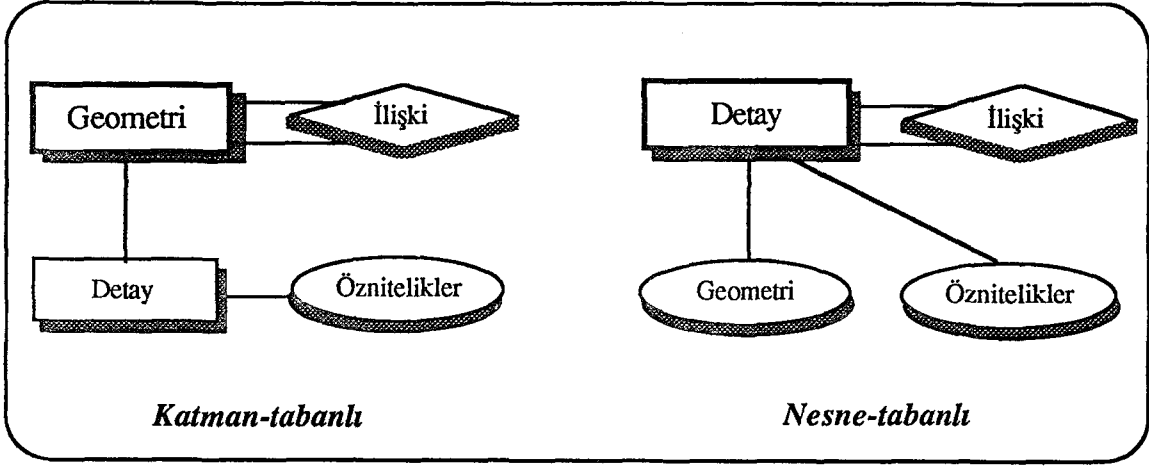
Kısım 1.2 de değinildiği gibi, konumsal veri için çok çeşitli konumsal veri modelleri kullanılmaktadır. Genel olarak, veri modeli açısından bakıldığında ND model alışlagelmiş katman-tabanlı modellere bir takım ölçütler bazında önemli üstünlükler sağlamaktadır. Bu üstünlük özellikle genişletilebilirlik açısından son derece belirgindir. Ancak bugün kullanımda olan çoğu CBS nin katman-tabanlı bir modele dayandığı ve pek çok transfer verisinin halen bu modda geldiği de bir gerçektir. Teorik olarak, DF nin veri modelinin katman-tabanlı olması, taraflardan birinin ya da her ikisinin nesne-tabanlı olduğu bir veri değişimini güçleştirecektir. Benzer şekilde, DF nin veri modelinin nesne-tabanlı olması, kayıt-tabanlı tarafların yer aldığı bir veri değişiminde ilave yük getirecektir. O nedenle DF farklı modellere dayalı veri grupları arasında çevirileri olanaklı kılacak kadar genel, farklı gereksinimlere sahip özellikli transferler için uyarlanabilecek kadar esnek, ve konumsal veri yönetimi gibi çok dinamik bir alanda sürekli değişen gereksinimler karşısında genişletilebilir olmalıdır. Bu koşulları sağlama derecesi DF nin "standart" olarak kabul görmesinde çok önemli bir rol oynayacaktır.

2.2.1. Veri Modeli Tipi

Konumsal veri modelleri hakkında ayrıntılı bilgi kısım 1.2 de verildiği için burada yalnızca, veri değişimi açısından genel bir değerlendirme yapılacaktır. Konumsal veri modelleri "nesne-tabanlı" ve "kayıt-tabanlı" olmak üzere iki gruba ayrılabilir. Nesne-tabanlı modeller, veri tabanının konusunu oluşturan gerçeği çeşitli tipte "nesne"lerin bir bütünü olarak görür. Nesneler konumsal ve konumsal olmayan öznitelikleri ve diğer nesnelerle olan ilişkileri ile tanımlıdır. Buna karşılık kayıt-tabanlı modellerde, nesne-tabanlı modellerde tanımlandığı şekliyle bir "nesne" kavramı mevcut değildir. Onun yerine gerçek sıfır, bir, ve iki boyutlu temel geometrik elemanlar—nokta, çizgi, poligon—yardımıyla modellenir. Bu geometrik elemanların birbirleriyle olan konumsal ilişkileri topolojik olarak tanımlanır [42]. Geometrik elemanlar veri tabanında, gerçekte gösterim ettikleri detay bilgileri—konumsal olmayan öznitelikler—ile ilişkilendirilir. Nesne-tabanlı modeller veri tabanı yapısını, kayıt-tabanlı modellerden daha yüksek bir düzeyde tanımlar. Nesneler arasındaki ilişkilerin, kayıt-tabanlı modellerde nesneleri gösterim eden geometrik elemanlar, nesne-tabanlı modellerde ise doğrudan nesneler düzeyinde gösterim edilmesi bunun bir göstergesidir (Şekil 24).

Bugün kullanımda olan çoğu CBS'nin veri modeli [42], [44] katman-tabanlıdır. Bu sistemlerde konumsal olmayan veri bir İVTYS, konumsal veri de özel veri yapıları yardımıyla yönetilir. Bunun nedeni, İVTYS lerin konumsal veri yönetimindeki

yetersizliđidir. Gerçekleřtirimde temel birim, öznitelik bilgilerine göre düzenlenmiř (Örn. yalnızca karayollarını içeren) "katman"lar dır. Kayıt-tabanlı modeller bazen "geometri-tabanlı" [16], bazen de "katman-tabanlı" [43] modeller olarak anılır.



Şekil 24. Nesne-tabanlı ve katman-tabanlı konumsal veri modelleri.

Konumsal veri deđiřiminde taraflar ve DF, nesne-tabanlı ya da kayıt-tabanlı bir modele sahip olabilir. Veri deđiřiminin gerçekleřebilmesi için her řeyden önce, modeller arasında çeviri olanaklı olmalıdır. Bugün kullanımda olan katman-tabanlı ve nesne-tabanlı sistemlerin çođunluđu için geçiř olanaklıdır [15], [16]. Çeviri iřleminin ne kadar kolay olacađına dair bir kural koymak güçtür. Ancak oldukça genel bir kural olarak, çeviri iřleminin iki modelin "anlamsal" yakınlıđu ölçüsünde kolay olacađı söylenebilir. Buna göre, DF nin veri modelinin katman-tabanlı olması, taraflardan birinin ya da her ikisinin nesne-tabanlı olduđu bir veri deđiřimini güçleřtirecektir. Benzer řekilde, DF nin veri modelinin nesne-tabanlı olması, kayıt-tabanlı tarafların yer aldıđu bir veri deđiřiminde gereksiz bir yük getirecektir. Bu nedenle DF nin hangi tip veri modeline sahip olması gerektiđini söylemek güçtür. Ancak genel olarak, DF nin her iki tip modele dayalı veri grupları arasında çeviriyi olanaklı ve pratik kılacak genellik ve esnekliđu sađlaması gerekir.

2.1.2 Raster Veri

Konumsal veri için vektör ya da raster veri yapıları kullanılabilir. Vektör yapıda temel birimler nokta, çizgi ve poligonlardır. Nokta, çizgi, ve poligonların Cebrik Topoloji'de tanımlı karřılıkları sırasıyla sıfır, bir ve iki boyutlu "göze" lerdir. Bunlar "0-göze", "1-göze", ve "2-göze" olarak anılır. Cebirsel Topoloji'de her bir göze daha yüksek boyutlu gözelere sınır oluřturur; bir çizginin iki poligonu birbirinden ayırması gibi. Vektör veri yapısında, gözelerin konumu mutlak—x,y,z koordinatlarıyla—ve/veya bađlı—topolojik—olarak ifade edilebilir.

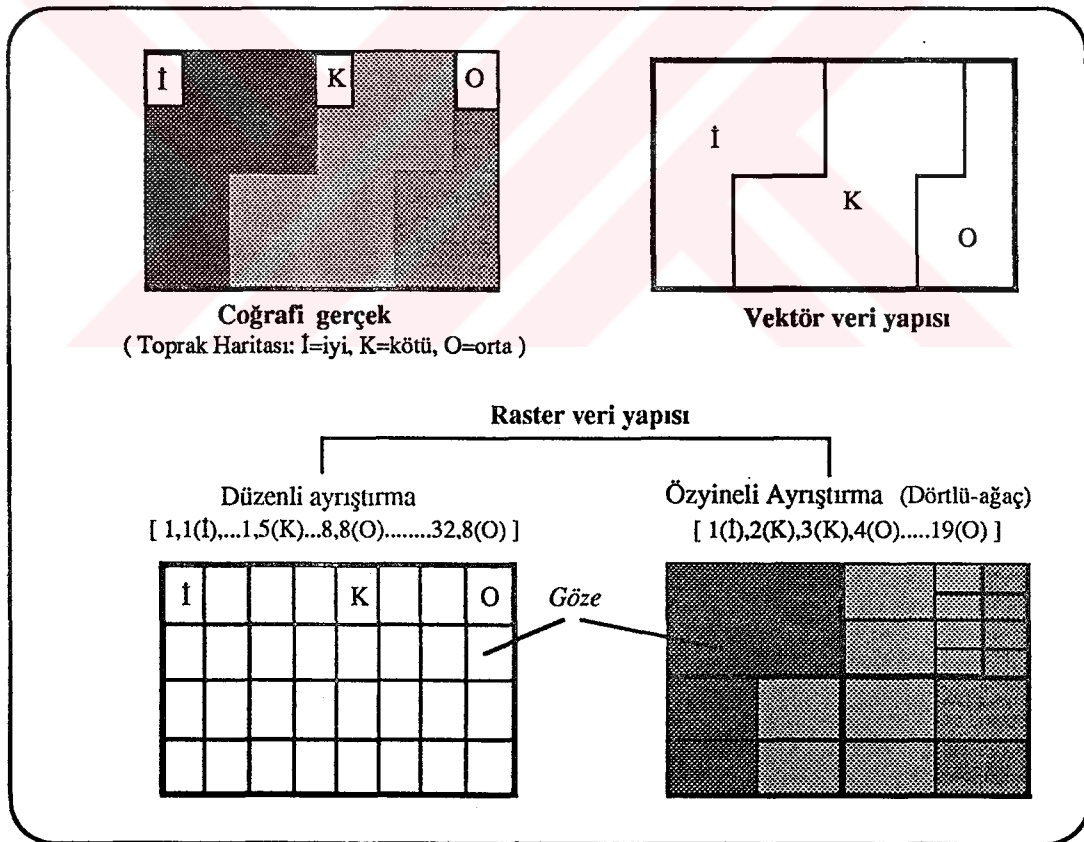
Raster yapı, coğrafi gerçeğin konumsal olarak düzenli boyut ve biçimdeki gözelerle ayrılması ve her göze karşılık gelen karakteristiklerin kaydedilmesi ilkesine dayanır (Şekil 25). "Karakteristikler" spektral yansıma, toprak cinsi, ya da yükseklik olabilir. Gözeler kare, dikdörtgen, üçgen ve diğer şekillerde olabilir. Ancak en çok kullanılan kare şeklinde bölümlenir. Bölümlenmenin çözünürlüğü yani göze boyutu teknolojik sınırı aşmamakla birlikte, gösterim konusunun karakteristiklerine ve eldeki uygulama gereksinimlerine bağlı olarak belirlenebilir. Göze boyutunun bilgisayarda gösterim edilebilen ya da sayısal kayıt ortamında kaydedilebilen en küçük birim olması durumunda, "göze" yerine çoğunlukla "pixel" (picture element) terimi kullanılır. Raster modunda gösterim edilen veri, çoğunlukla "raster veri" bazen de "matris veri", bölümlenme ile oluşan yapı ise "raster" ya da "grid" olarak anılır.

Raster veri bugün konumsal veri yönetimi alanında çok önemli bir yer tutmaktadır. Bunun üç temel nedeni vardır. Birincisi günümüzde artık konumsal verinin büyük bir bölümü raster veri modunda gelmektedir. Buna örnek olarak, uydulardan alınan uzaktan algılama verisi ile, kullanımı gittikçe yaygınlaşan tarayıcılardan alınan çıktı veri verilebilir. İkincisi bazı konumsal analizler için raster modun daha uygun olmasıdır [105]. Üçüncü olarak, görüntü ekranları ve çiziciler gibi çıktı aygıtlarının büyük bir çoğunluğu raster modunda çalışmaktadır. Bu durum, CBS ler dünyasında 1980 lerin başlıca ilgi odaklarından biri olan, vektör ve raster tabanlı sistemlerden hangisinin daha üstün olduğu tartışmasını da sonuçlandırmıştır; Bugün artık bir CBS nin her iki modda da çalışabilmesi bir zorunluluk olarak görülmekte ve bir moddan diğerine geçişin son kullanıcıya hissettirilmeden gerçekleştirilmesi "bütünleşik sistemler" in başlıca gereklerinden biri olarak kabul edilmektedir [15], [106].

Raster veri çok çeşitli şekillerde organize edilebilir. Genellikle, göze koordinatları mutlak (x, y, z) olarak saklanmayıp, belli bir indeksleme yapısındaki değerinden elde edilir. İndeksleme yapısı dizi ya da ağaç yapıları olabilir. Örneğin dizi kullanılması durumunda göze koordinatları, gözenin satır ve sütun numaraları ile ifade edilebilir. Gösterim edilen karakteristiklerle gözelerin ilişkilendirilmesinde amaç, en az depolama alanı ile en yüksek verimdir. Eğer raster gösterime konu olan tek bir karakteristik (örn. toprak cinsi) ise, veri organizasyonu nispeten basittir. Eğer birden fazla karakteristiğin (örn. LANDSAT MSS görüntüsü) gösterimi söz konusu ise, veri organizasyonu için bir takım seçenekler mevcuttur [105]. Bunlara iki örnek vermek gerekirse, bir yöntem karakteristiklerin gözelerle göre organize edilmesidir. Bunu esas alan bir kodlamada, her göze için göze pozisyonunu ve göze karşılık gelen karakteristiklerin değerleri yer alır. Bu LANDSAT MSS örneğine uygulandığında, göze pozisyonu pixelin bulunduğu satır ve sütuna, karakteristik değerleri de

her dört banda ait spektral yansıma (gri tonları) değerlerine karşılık gelecektir. Diğer bir yöntem, her bir karakteristik için ayrı bir katman düzenlemektir. Bu LANDSAT MSS örneği için dört ayrı katman anlamına gelir. Dolayısıyla bir katmandaki her pixel için yalnızca bir gri tonu değeri bulunacaktır.

Raster veri yapısında, eldeki projeye ve gösterimi yapılacak karakteristiğin dağılımına göre eniyileştirme teknikleri uygulanabilir. Buna göre, aynı bir bölümlene içerisinde değişken çözünürlük uygulanabilir. Diğer bir anlatımla, bazı gözeler kendi içerisinde ayrıca bölümlenebilir (Şekil 25). Bölümleme aynı bir göze içerisinde homojen karakteristik değerlerine ulaşıncaya kadar devam eder. Bu çoğunlukla, "özyineli ayrıştırma" olarak anılır. Bunun sonucunda hiyerarşik bir yapı oluşur. Dörtlü ağaç yapısı [45] bunun en belirgin örneklerinden biridir (bkz. 1.2). Dörtlü ağaç yapısı raster verinin daha özlü bir şekilde gösterim edilmesini sağlar. Bunun yanında raster verinin kodlanmasında çeşitli sıkıştırma teknikleri uygulanabilir [105].



Şekil 25. Raster veri yapısı.

Sonuç olarak, raster veri için çok değişik kodlama ve sıkıştırma teknikleri uygulanması sonucu, raster veri bugün çok çeşitli formatlarda gelebilmektedir. Bu formatlardan en çok bilinen bazıları TIFF, Landsat TM, SPOT, GIF, RLE, JPEG, BMP, PICT, TARGA dır.

Dolayısıyla, DF öncelekle raster veri için gerekli desteği sunmalı ve farklı raster formatlarında veri gösterimine izin vermelidir.

2.1.3 Kartografik Gösterim

Kartografya'nın bugün yaygın olarak kabul edilmiş bir tanımı mevcut değildir [4]. [40]'a göre Kartografya, coğrafik verinin gösterimi ve gösterimi ile ilgilenir. [107]'ye göre Kartografya, coğrafi gerçeğin çok yönlü bir modeli olarak kabul edilebilecek bir konumsal veri tabanı merkezli bir bilgi transferi işlemidir. Yazara göre, Kartografya'nın temel ilgi alanı coğrafik verinin kullanıcıya sunuluşudur. Kartografya'nın tanımı, nelerin "kartografik gösterim" kapsamına girdiğinin belirlenmesi ve burada konu dışında olmakla birlikte, CBS ler kapsamında Kartografya'nın yeni rolünün ne olması gerektiği açısından önem taşımaktadır.

Konumsal veri kartografik olarak çok çeşitli şekillerde gösterilebilir. Örnek olarak bir akarsu ismi alınsın. Bir gösterim $x, y, açı, uzunluk, isim$ şeklinde olabilir. Burada x, y ismin başlangıç koordinatları, $açı$ ismin yöneltme açısı, $uzunluk$ ismi oluşturan karakterlerin sayısıdır. Daha ayrıntılı bir gösterim ise $uzunluk, x, y, açı, karakter, x, y, açı, karakter...$ şeklinde olabilir. İkinci gösterimde ismin her bir karakteri için konum ve açı bilgisi yer almaktadır. Bu gösterim, ismin eğri bir çizgiyi izleyebilmesini sağlar. Böylece akarsu ismi, akarsu çizgisi üzerinde, birinci gösterimin sağladığından daha estetik bir şekilde gösterilebilir. Bu bakımdan, DF'nin farklı kartografik gösterimlere olanak tanınması gerekir.

Bir haritada nokta, çizgi, ve poligon detaylar ve isimler çeşitli şekillerde gösterilebilir. Örneğin kentler nüfuslarına göre değişik yarıçaplı dairelerle, otoyollar 1 mm kalınlığında, kırmızı renkte çizgilerle, göller içi mavi çizgilerle taranmış poligonlarla gösterilebilir. İşaretler belirli parametrelerle karakterize edilirler. Bunlar işaretin konumu, boyutları, fontu, ve rengi ile ilgili parametrelerdir. İşaretler genellikle "işaret dosyaları" yardımıyla organize edilir. Bir işaret dosyasında işaretin kimlik kodu, işaretün şekli ve işaretün hangi detaylar için kullanılacağını gösteren "detay kodu" (DK) gibi bilgiler bulunur. Aynen konumsal veri değişiminde olduğu gibi, işaretlendirme açısından da standartlaşma gereklidir. Örneğin elektronik yönlendirme haritaları açısından ortak işaretlerin kullanılması son derece önemlidir.

İşaret dosyalarının oluşturulması oldukça zaman alıcı bir iştir ve genellikle işaret ve/veya karakterlerin sayısallaştırılmasını içerir [37]. Bu bakımdan işaretlerin transfer edilmesi gereklidir ve alıcı sistem için çok önemli bir kazanç oluşturacaktır. Ancak, alıcı sistemin aynı detaylar için sunucudan farklı işaretler kullanması durumunda, işaret transferi anlamsızdır.

Bu sorun "işaret kitaplıkları" kullanılarak çözülebilir. Bu yolla, transfer veri grubu alıcı sistem işaretlerini içeren bir işaret kitaplığına gönderimde bulunabilir. Alıcı sistemin sözkonusu işaret kitaplığına sahip olmaması durumunda, işaret kitaplığının da transfer veri grubu içerisinde yer alması gerekir. Sonuç olarak, DF'nin işaret transferini desteklemesi aranan bir özelliktir.

Konumsal bilginin son kullanıcıya sunuluş biçimini belirleyen etkenlerden biri olması açısından, harita projeksiyonları da kartografik gösterim kapsamında düşünülebilir. Desteklenen projeksiyonların sayısı bir DF'den diğerine değişir. Bir DF'nin çok sayıda projeksiyon için desteğinin bulunması, aranan bir özelliktir. Ancak yaygın olarak kullanılan projeksiyonlar için desteği bulunan bir DF'nin, projeksiyon desteğinin genişletilmesi zor bir işlem olmadığı için, bu çok önemli bir ölçüt değildir. Çoğu DF, çok yaygın olarak kullanılan projeksiyonları (örn. Mercator, UTM, Lambert konform konik) desteklemektedir. Bir DF'nin bir projeksiyonu desteklemesi için o projeksiyona ait projeksiyon kodu ("UTM" gibi), ve merkezi (orta) meridyen, teğet meridyen, teğet paralel, ölçek faktörü, değiştirilmiş koordinat merkezi gibi projeksiyon parametrelerinin transfer veri grubu içinde kodlanmasına olanak tanınması gerekir.

2.1.4 Koordinatlar

Koordinatlar 3 boyutlu yersel ya da 2 boyutlu düzlem koordinatlar olarak ifade edilebilir. Yersel koordinatlar ya üç boyutlu bir dik koordinat sisteminde (x, y, z), ya da coğrafi koordinat sisteminde (enlem, boylam, yükseklik) olabilir. Düzlem koordinatlar basit düzlem dik koordinat sisteminde, kutupsal bir sistemde, bir projeksiyon sisteminde, ya da bir raster gridinde ifade edilebilir. Her koordinat sisteminin bir referans sistemi vardır. Referans sistemi yatay ve düşey konum için referans yüzlerini ve koordinat merkezini belirler. Yatay referans yüzeyi yatay konum için geçerli olan dengeleme sistemi ve referans elipsoidi ile tanımlıdır. Benzer şekilde düşey referans yüzeyini belirtmek için, düşey dengeleme sistemi ve düşey datum gibi parametreler kullanılır. Farklı referans sistemlerindeki koordinatların birbirlerine dönüştürülebilmesi için referans sistemi parametreleri bilinmelidir. Ulusal ve Uluslar arası bazda çok çeşitli koordinat ve referans sistemleri sözkonusu olabilir. Değişik koordinat sistemlerinde koordinatların belirtilebilmesi ve bu koordinat sistemleri arasında koordinat dönüşümlerinin gerçekleştirilebilmesi için DF'nin farklı koordinat, projeksiyon, ve referans sistemlerini tanıması gerekir.

Genellikle alışlagelmiş DF'ler ancak 2 ya da 3 boyutlu koordinatlara izin verirler. Dördüncü boyut, zamansal veriye olan gereksinim nedeniyle yakın zamanda önem

kazanmıştır (bkz. 2.11). O nedenle DF'nin zamanın da bir koordinat olarak kodlanmasına olanak tanınması gerekir.

Konumsal veri değişiminde bulunan tarafların koordinatlar için farklı çözünürlükler kullanmaları problemlere neden olabilir. Örneğin kesişen iki çizgi transferden sonra bu koşulu sağlamayabilir. Genel olarak "çözünürlük", gösterim edilebilen en küçük birim anlamına gelir. Koordinatların gösterimi için kullanılacak anlamlı basamak sayısı ile doğrudan ilişkilidir. Örneğin koordinatlar için 32 bit lik tamsayı kullanılması durumunda saklanabilecek en büyük değer, 10 anlamlı basamak ile, 4 294 967 295 dir. Bu, örneğin 4000 km lik bir koordinat değerinin 1mm çözünürlükle gösterimi için yeterlidir. Bu bakımdan, DF koordinat kodlama çözünürlüğü için çeşitli seçenekler sunmalı ve istenen duyarlılıkta koordinat kodlamaya olanak tanınmalıdır. Bazı alışlagelmiş DF'ler, kullandıkları kodlama yöntemleriyle anlamlı basamak sayısını pratik olarak sınırsız yapmaktadırlar [12].

2.1.5 Topoloji-Geometri Ayırımı

Topoloji, geometrik şekillerin burkulma, büyüme ve küçülme gibi, tamamen bozulma dışındaki çeşitli şekil değişiklikleri karşısında hiç değişmeyen özellikleri ile ilgilenir [108]. Sözkonusu şekil değişiklikleri kauçuk bir yüzeyin yırtılmadan kıvrılması, uzatılması vs. benzetilebilir.

Bir haritada iki tür bilgi bulunur. Bunlar "geometrik" bilgi ve geometrik olmayan, "nitelik" bilgisidir. Örnek olarak, bir karayolunu gösterim eden çizgi koordinatları geometrik bilgisidir. Bu karayolunun yapım yılı ve mevcut durumu gibi bilgiler ise nitelik bilgileridir. Geometrik bilgilerin ise, bir "metrik" ve bir de "topolojik" boyutu vardır [109], [110]. Bununla birlikte, "geometri" sözcüğü ile genellikle metrik boyut kastedilir [29]. Metrik bilgiler dönüşümler altında değişikliğe uğrayan bilgilerdir. Örneğin bir karayolu çizgisinin koordinatları farklı bir koordinat sistemine dönüştürüldüğünde değişecektir. Topolojik bilgiler ise dönüşümlerden etkilenmeyen, bir parselin diğer bir parselle komşu olması, noktaların aynı bir çizgi üzerinde bulunması, iki çizginin bir noktada kesişmesi gibi bilgilerdir. Örneğin birbirine komşu olan iki parselin bu topolojik ilişkileri, bir benzerlik dönüşümü sonucunda bozulmayacaktır.

Topolojik ilişkilerin bir CBS veri tabanında gösterim edilmesinin ana amacı, CBS nin analiz gücünü artırmaktır [37], [42]. Topolojik ilişkiler "topolojik veri yapıları" altında veri tabanında tutulur [37], [42]. Topolojik ilişkilerin hesaplanması "topoloji oluşturma" olarak adlandırılır [37]. Her ne kadar topolojik ilişkiler metrik bilgilerden elde edilebilirse de, bu ilişkilerin önceden derlenerek veri tabanına konması analiz hızını önemli ölçüde artırır. Buna karşılık, topolojik ilişkilerin ilk başta hesaplanarak veri tabanına konması ve değişiklikler

nedeniyle sürekli olarak yenilenmesi gerekir. Ayrıca topolojik ilişkilerin veri tabanında tutulması depolama alanı gereksinimini önemli ölçüde artırır. Bu bakımdan topoloji gereksinimi olmayan uygulamalar için topoloji bileşeninin iptal edilmesi—topolojinin oluşturulmaması—ve gerektiğinde tekrar devreye sokulması, günümüz CBS'lerinin aranan bir özelliği olmuştur.

Topolojik veri transferinin gerekliliği, tartışmalı bir konudur [12]. Topoloji transferi iki bakımdan gereksiz görülmektedir. Birincisi, topoloji alıcı sistemde yeniden oluşturulabilir. İkincisi, DF nin topolojik veri yapısının sunucu ve alıcı sistemlerinkinden oldukça farklı olması durumunda, topolojinin alıcı sistemde yeniden oluşturulması çok daha pratik olabilir. Buna karşılık topoloji transferini destekleyen etmenler vardır. Öncelikle, topoloji oluşturma zaman alıcı bir işlemdir. İkinci olarak, alıcı sistemin topoloji oluşturma yeteneği bulunmayabilir. Son olarak, topoloji veri transferinden sonra verinin geometrik tutarlılığını doğrulamak için kullanılabilir. Alıcı sistemin daha düşük bir çözünürlüğe sahip olması, geometrik tutarsızlık doğurabilir. Sonuç olarak her halde, DF nin topoloji desteğinin bulunması gerekir. Bununla birlikte, alıcı sistem topolojik veriye gereksinim duymayabilir ya da yalnızca metrik veriyi transfer etmek isteyebilir. Bu örneğin, alıcı sistemin transfer ettiği veriyi yalnızca harita üretiminde kullanması ya da topolojiyi kendi içerisinde ayrıca oluşturmak istemesi durumunda sözkonusu olabilir. Bu nedenle DF'nin geometrik veri transferi yanında, yalnızca metrik ya da topolojik veriyi transfer edebilme seçeneğini de sunması gerekir.

2.1.6 Karmaşık Nesneler

Nesneler "basit" ya da "karmaşık" olabilirler. Karmaşık nesneler daha basit nesneler içerirler. Örneğin bir poligon, çizgilerden çizgiler de noktalardan oluşur. Burada poligon karmaşık bir nesnedir. Bununla birlikte, çizgi ve hatta nokta da kendi içerisinde karmaşık birer nesnedir. Çünkü bir nokta kabaca, her biri basit birer nesne olan x , y , z koordinatlarından oluşur. Bir karmaşık nesne ile onu oluşturan eleman nesneler arasındaki "oluşum" ilişkisi iki farklı türde olabilir [35], [36]. Birinci türde, eleman nesnenin varoluşu karmaşık nesnenin varoluşuna bağlıdır. Aynı bir noktanın yalnızca bir poligona ait olduğu bir harita için, poligon örneği bu gruba girer. Çünkü bir poligonun geçersiz kılınması ile poligon sınırını oluşturan noktaların bir anlamı kalmayacaktır. İkinci türde böyle bir durum sözkonusu değildir. Bir karayolu_ağı ile onu oluşturan karayolu nesneleri arasında bu tür bir oluşum ilişkisi vardır. Çünkü karayolu_ağı varlığının geçersiz kılınması, onu oluşturan karayollarının da geçersiz kılmaz. Çünkü her bir karayolunun kendi başına da bir anlamı vardır.

Farklı tipte varlıkların yeni tipte ve daha yüksek düzeyde bir varlık oluşturmak üzere bir araya getirilmesine "birleştirme", bunun tersi işleme de "ayırıştırma" adı verilir (bkz. 1.2). Bir CBS nin birleştirme ve ayırıştırma mekanizmalarını desteklemesi, yani bu işlemler için gerekli yazılımları yapısında bulundurması çok önemli bir ayrıcalıktır. Öte yandan, karmaşık nesnelerin tek bir birim—nesne—olarak işlem görmesi, CBS kullanıcılarına uygulamalarını daha yüksek bir düzeyde modelleyebilme ve sorgulamalarını daha yüksek bir düzeyde yapabilme olanağı tanır. Bu da programcının ve uygulama programlarının yükünü çok önemli ölçüde hafifletir.

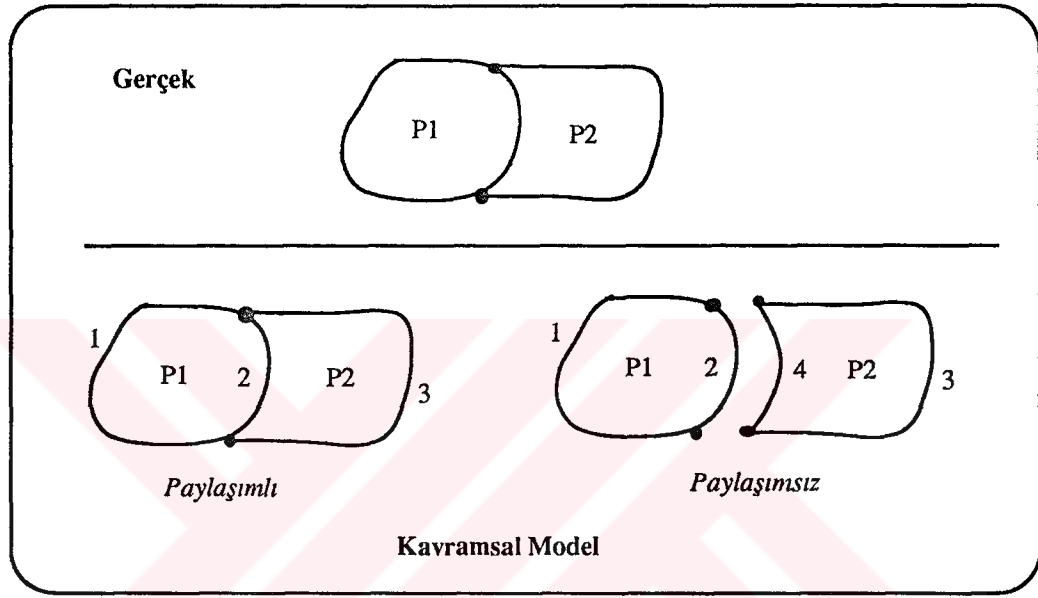
CBS lerde en çok kullanılmış veri modeli olan İlişkisel modelin, karmaşık nesne desteği oldukça zayıfır. Karmaşık bir nesnenin ilişkisel bir veri tabanındaki bir gösteriminde, eleman nesneler çeşitli ilişkiler üzerine dağılmış durumdadır. Çünkü ilişkisel veri tabanı tasarımında veri tabanının konusunu oluşturan gerçek, çeşitli ilişkiler üzerine aktarılacak durumdadır (bkz. 1.2). En karmaşık veri tipi "İlişki" dir. Bir ilişki kayıtlardan, kayıtlar ise alanlardan oluşur. Bir alanda bulunabilecek değerler ise sayı, karakter, tarih gibi basit veri tipinde olabilirler. Dolayısıyla karmaşık varlıkların ilişkiler üzerine aktarılmasında "anlamsal kayıplar" oluşur. Bu da, yitirilen anlamın uygulama programlarında bir şekilde yeniden kazanılmasını gerektirdiğinden, uygulama programları ve dolayısıyla programcının yükünü artırır. Karayolu-ağı örneğine dönülecek olursa, bu nesnenin bir Soyut Veri Tipi (SVT) olarak tanımlanması ile konumsal analizler—örneğin iki yer arasında en kısa yolun bulunması—tipik bir İVTYS dekinden çok daha yüksek bir düzeyde gerçekleştirilebilecektir.

Bir DF nin karmaşık nesne desteğinin olması, karmaşık nesnelerin doğrudan değişimini olanaklı kılması anlamına gelir. Bu da DF nin veri modelinin karmaşık nesne kavramını içermesi ve karmaşık nesnelerin aynen basit nesneler gibi tek bir birim olarak işlem görmesini sağlaması demektir. Yukarıdaki açıklamalardan anlaşılacağı üzere, DF'nin karmaşık nesne desteğinin bulunması çok büyük bir önem taşımaktadır. Veri modeli desteğine paralel olarak, DF nin gerçekleştiriminde birleştirme/ayırıştırma yapıları desteklenmelidir.

2.1.7 Nesne Paylaşımı

"Nesne paylaşımı", birden fazla nesnenin veri tekrarına yol açmaksızın aynı bir nesneyi paylaşmalarıdır. Örnek olarak iki poligon arasındaki ortak çizgi parçası alınsın (Şekil 26). Nesne paylaşımının uygulanmaması durumunda ortak çizgi parçası her iki poligon için de ayrıca kodlanmalıdır (2 ve 4 nolu çizgi parçaları olarak). Nesne paylaşımının uygulanması durumunda ise, ortak çizgi yalnızca bir kez kodlanacak (2 nolu çizgi parçası olarak) ve her iki poligon da 2 nolu çizgi parçasına işaret edecektir.

Depolama alanı açısından nesne paylaşımı, özellikle çoğunluklu olarak alansal veri söz konusu olduğunda önemli bir kazanç sağlayacaktır. Çünkü alan (poligon) sınırları iki poligon için de ortaktır. Buna ek olarak, alansal veri ile çizgisel veri arasında da paylaşım söz konusu olabilir. Örneğin parsel sınırları ile akarsu ya da yol sınırları çakışabilir. Depolama alanı yanında nesne paylaşımı, güncelleştirme açısından da önemli kolaylıklar sağlayacaktır. Söz konusu kazançlar, özellikle geniş veri tabanları için son derece anlamlıdır.



Şekil 26. Nesne paylaşımı kavramı.

Bugün bir CBS nin nesne paylaşımını desteklemesi bir ayrıcalık olarak görülmektedir. Konumsal veri değişiminde de nesne paylaşımının uygulanması transfer veri grubunun boyutunu küçültecek ve dolayısıyla transfer zamanını azaltacaktır. Bu nedenle, DF veri modelinin nesne paylaşımını desteklemesi gerekir.

2.1.8 Meta Veri

Meta veri, veri hakkında veri demektir. Meta veri "veri kalitesi"nin, yani verinin belli bir uygulama için uygunluğunun ya da yeterliğinin, belirlenebilmesi için gereklidir. Meta veri verinin doğruluğu, tamlığı (bir takım bilgilerin eksik olup olmadığı), güncelliği, hangi kaynaktan ve nasıl elde edildiği (hangi işlemlerden geçtiği vs.), verinin referans yüzeyi gibi bilgiler içerir. Veri kalitesi uzun zamandan beri tartışılan ve çoğu kere gözardı edilen bir kavram olmakla birlikte, özellikle bütünleşik sistemlerin gündeme gelmesi ile bir hayli önem kazanmış ve konumsal veri değişimine yeni bir boyut daha eklemiştir. Bu nedenle son zamanlarda geliştirilen DF lerde veri kalitesi desteği aranmaktadır [29].

Veri kalitesi ile ilgili iki temel problem sözkonusudur. Bunlardan biri veri kalitesinin nasıl ölçüleceği, diğeri ise nasıl ifade edileceğidir [12]. Çünkü, veri kalitesi göreceli bir kavramdır; bir kullanıcı için bir veri grubunu "iyi kalite" olarak tanımlayan şeyler, diğeri bir kullanıcı için tamamen uygunsuz olabilir. Bunun sebebi veri kalitesinin büyük ölçüde uygulama bağımlı olmasındandır. Bu nedenle veri kalitesinin standart bir şekilde belirtilmesinde sorunlar ortaya çıkmaktadır [11], [111], [112]. Bu bakımdan meta verinin kantitatif olarak belirtilmesi yerine, "veri kalitesi raporu" şeklinde belirtilmesi daha uygun olabilir [112].

Bilgisayar Destekli Tasarım (BDT) standartları olarak kabul edilen SIF ve DXF gibi formatlar meta veri belirtmeye olanak tanımazken, meta verinin her geçen gün artan önemi sonucu, son zamanlarda geliştirilen DF'lerde meta veri desteği bulunmaktadır. Bir DF in meta veri desteğinin geniş olması, yani meta veri olarak kullandığı etmen sayısının fazla olması, transfer veri grubunun uygunluğunun belirlenmesindeki duyarlılığı artıracaktır. Ancak çok sayıda meta veri değerinin hem üretilmesi ve hem de değerlendirmesi kolay bir iş değildir. Bu nedenle DF, kullanıcıya en uygun meta veri kapsamını belirtme olanağı tanımalıdır.

Bir veri grubu için meta veri, ya veri grubunun bütünü için genel olarak ya da tek tek nesne bazında belirtilebilir. Bazı DF ler yalnızca genel olarak meta veri belirtmeye izin verirken, özellikle daha yeni DF lerde her iki seçenek de bulunmaktadır. Geleceğin veri değişim şekli olarak görülen nesne-tabanlı veri değişimi için nesne bazında meta veri belirtme gereklidir. Sonuç olarak DF, transfer veri grubunun tamamı ya da tek tek nesneler bazında, değişen ayrıntı düzeyinde meta veri belirtmeye olanak tanımalıdır.

2.1.9 Nesne Bazında Veri Değişimi

"Nesne-tabanlı" veri değişimi ile, değişimin nesneler bazında gerçekleştirilmesi kastedilmektedir. Geleneksel olarak, veri değişimi bütün bir dosyanın—transfer veri grubu—transferi olarak uygulanmıştır. Bugün kaçınılmaz hale gelen interoperabilite, nesne bazında veri değişimini gerektirmektedir. Çok genel olarak interoperabilite, farklı uygulama ya da işlemlerin aynı veriyi paylaşabilmeleridir [15]. Böyle bir ortamda veri değişimi farklı işlemler arasında koşum zamanında gerçekleşir. Gerçekte bu, ulaşılması öyle kolay bir hedef değildir. Temel problem, farklı uygulamaların aynı veriyi oldukça farklı perspektiflerden görmeleridir. Bu farklılığın klasik örneği vektör-raster veri yapılarıdır. Son bir kaç yıldır da, ilişkisel ve ND modeller arasındaki farklar gündeme gelmiştir. Özetle sözkonusu olan, kayıt-

tabanlı ve nesne-tabanlı modeller ve bu modellere dayalı farklı veri yapıları (raster-vektör gibi) arasındaki farklılıkların uyuşturulmasıdır.

İdeal olarak, bir kullanıcı bütün bir paftayı transfer edebileceği gibi yalnızca bir karayolunu ya da bir kent sınırını da transfer edebilmelidir. Örneğin bir karayolu projesi için yalnızca farklı dosyalarda bulunan karayolu verisi ve topoğrafik verinin birleştirilmesi gerekebilir. Bu durumda, tipik bir veri değişimi işleminin diğer problemleri bir yana, yalnızca ilgili karayolu yerine, karayolunu içinde bulunduran dosyanın bütününün transferi anlamsızdır. Bu, insan ve bilgisayar kaynaklarının israfı demektir. O nedenle DF nin nesne bazında veri değişimini ve transfer anında nesne elemanlarının—öznitelikler, meta veri, ilişkiler—bir araya getirilmesini desteklemelidir. Bu açıdan nesne-bazında veri değişimi, formatın gerçekleştirimi ile de ilgili olmakla birlikte, bu ölçüt ile aranan, gerekli veri modeli desteğidir. Bu bakımdan nesne-tabanlı DF ler daha avantajlıdır. Çünkü nesne-tabanlı modeller nesnelerin tek bir birim olarak doğrudan gösterimi için gerekli yapıyı sağlamaktadır. Katman-tabanlı bir modele dayalı bir DF ile nesne bazında veri değişimi, alışlagelmiş programlama dili ile ND programlamaya benzetilebilir.

2.1.10 Zamansal Veri

"Zamansal veri", "zamansal veri tabanları", ve "zamansal CBS" de, "meta veri" gibi son yıllarda sıkça duyulan sözcükler olmuşlardır. Geleneksel VTYS lerde olduğu gibi, CBS ler için de zaman boyutu uzun süre ihmal edilmiş, bu konu ile ilgili çalışmalar ancak 1980'lerden sonra hız kazanabilmiştir [113]. Zaman boyutunun CBS kapsamına alınması bugün hala çözümlenmiş değildir ve CBS ler alanındaki başlıca araştırma konularından birini oluşturmaktadır [114], [115], [116].

Bugün kullanılan CBS'lerde veri tabanı, gösterim ettiği gerçeğin yalnızca en son durumunu yansıtır. Çünkü gerçeğin son durumu sürekli olarak bir önceki durum üzerine kaydedilir, ve önceki durum silinir [113], [117]. Bu da geçmişteki durum hakkında bilgi gerektiren, gelecek için kestirimlere gereksinim duyan pek çok CBS uygulaması için sorun yaratmaktadır. Kaynak yönetiminde, şehir ve bölge planlamasında, arazi mülkiyeti yönetiminde, çevre izleme sistemlerinde konumsal analizler yanında, "zamansal analizler" de temel fonksiyonlardan biri olmak zorundadır. Örnek olarak orman kaynaklarının yönetiminde orman amenajman planlarının yapılması, bir hastalığın yayılmasının izlenmesi, ya da bir orman yangınının birtakım dinamiklere göre gelişiminin kestirilmesi zamansal analizler içerir. Örneğin orman yangınının saat 12 ve 14 deki durumu biliniyorken, ormanın zeminin yapısı, ağaç türleri, ve günün meteorolojik koşulları altında saat 20 deki durumunun ne olacağının kestirilmesi ve daha da ileriye giderek bunun bilgisayar ekranında

görselleştirilmesi gibi. Benzer şekilde, arazi mülkiyeti yönetiminde eski tapu kayıtlarına çoğunlukla gereksinim duyulur [117]. Bu nedenle geleneksel kadastral sistemlerde eski tapu kayıtları arşivlenir. Bu bakımdan, bilgisayar destekli sistemlerin geleneksel sistemlerin işlevlerini tamamiyle yerine getirebilmeleri için arşivleme yeteneğine de sahip olmaları gerekir.

Zamansal analizlerin yapılabilmesi veri tabanının gerçeğin farklı zamanlardaki durumunu yansıtabilmesi ile olanaklıdır. İşte "gerçeğin farklı zamanlardaki durumu", ya da diğer bir deyişle farklı zamanlara ait veri "zamansal veri" olarak anılır. Zamansal veri gereksinimi, arazi mülkiyeti yönetiminde olduğu gibi çok uzun bir zaman dilimi ya da orman yangını örneğindeki gibi çok kısa bir zaman dilimi için sözkonusu olabilir. Sorun, zamansal verinin veri tabanına nasıl sokulacağı, yani gerçeğin farklı zamanlardaki durumunun nasıl gösterim edileceğidir. Bunun için öncelikle konumsal veri modelleri gerekli desteği sağlamalıdır. Bildirinin kapsamı gereği burada zamansal verinin gösterimi ile ilgili kavram ve sorunlar üzerinde detaylı olarak durulmayacaktır. Yalnızca ilgili temel kavram ve sorunlara izleyen paragrafta kısaca değinilecektir. Bu konuda daha fazla bilgi için bu kısımdaki kaynaklara başvurulabilir.

Zaman içerisinde bir varlık ya yok olabilir ya da bir takım nitelikleri değişebilir. Örneğin bir parsel tamamiyle varlığını yitirebileceği gibi, konumu, alanı veya sahibi değişebilir. Her iki durum da zamansal veri modelleme kapsamındadır. Varlıkların zamana bağlı bu değişiklikleri kesikli ya da sürekli bir tarzda gerçekleştirebilir. Örneğin parsel değişiklikleri kesikli tarzda oluşan değişikliklerdir. Bir varlığın zaman içerisinde uğradığı değişiklikler onun diğer varlıklarla olan konumsal ilişkilerini etkileyebileceği gibi, "zamansal ilişki"lerini de etkileyebilir. İki varlık arasında bir zamansal ilişkiye örnek olarak, varlıklardan birinin varoluşunun diğerinin yokoluşuyla başlaması verilebilir. Zamansal ilişkilerin modellenmesi için "zamansal topoloji" önerilmektedir [35], [115], [117]. Zamansal topoloji, varlıkların zaman içerisindeki değişik durumlarının tek boyutlu ve tek yönlü bir "zaman çizgisi" üzerine izdüşürülebileceği ilkesine dayanır. Bu çizgi üzerindeki her bir olay ilgili varlıklar için bir göstergeç bulundurulur. Böylelikle, "Bu parsel üzerindeki binanın yapılmaya başlandığı tarihte civardaki diğer parseller üzerinde hangi binalar vardı?" gibi sorgulamalar yanıtlanabilir. Bütün sorun, gereksinim duyulacak zamansal analizleri olanaklı kılacak zamansal ilişkilerin belirlenmesinde ve bu ilişkilerin bir şekilde gösterim edilmesindedir.

Konumsal verinin modellenmesinde olduğu gibi, zamansal verinin modellenmesinde de geleneksel katman-tabanlı tekniklerle, NY tekniği karşı karşıyadır. Katman-tabanlı yaklaşımda her bir değişik durum—ya da her farklı zaman—için bir katman bulunur. ND

yaklaşımında ise katmanlar yerine nesneler vardır. Bir nesne tanımı hem nitelikleri ve hem de işlemleri kapsadığı için veri tabanındaki gerçeğin farklı zamanlardaki durumu, farklı nesnelerle gösterim edilebilir. Zamansal ilişkilerin nesneler düzeyinde gösterim edilmesi, zamansal analizlerin daha yüksek bir düzeyde yapılabilmesini sağlar. Bu iki farklı yaklaşımın bir değerlendirmesi [115] de bulunabilir.

Bir DF için zaman boyutu desteği, DF nin veri modelinin zamansal verinin modellenmesi için birtakım kavram ve yapıları kapsamaları ile ilişkilidir. Zamansal verinin CBS ler için önemi karşısında, bir DF nin zaman boyutu desteğinin bulunması başlıca aranan özelliklerden biri olmalıdır.

2.1.11 Multimedya

Diğer alanlarda olduğu gibi CBS ler alanında da multimedya uygulamaları, CBS lere yeni bir boyut daha eklemiştir. Multimedya ile örneğin, bir arazi parçası üzerinde üç boyutlu bir gezinti bir animasyon şeklinde kullanıcıya sunulabilmektedir. Bu animasyon, gezintinin bir motorsiklet, başka bir araç ile ya da bir füze uçuşu olmasına göre değişebilmektedir. Yine Multimedya ile örneğin bir karayolunun durumu bir ses kaydı olarak ifade edilebilecektir. Multimedya uygulama örnekleri çoğaltılabilir. Multimedya verisi olarak anılan "ses" ve "video" verisi çok çeşitli şekillerde gösterim edilebilmektedir. DF öncelikle multimedya veri tiplerini ve bu verinin farklı gösterimlerini desteklemelidir.

2.1.12 Genişletilebilirlik

Konumsal veri ile başa çıkmada, her ihtiyaca cevap verecek genel bir veri modeli yaklaşımının pratik bir çözüm olmadığı, bugün artık çok yaygın olarak kabul edilmektedir [118]. Bunun iki temel nedeni vardır. Birincisi, konumsal verinin orman mühendisinden arkeologa, çok geniş bir kullanıcı yelpazesine sahip olması, yani CBS kullanıcı gereksinim ve beklentilerin farklılığıdır. İkincisi, gereksinim ve beklentilerin zamana bağlı olarak sürekli değişmesidir.

DF ler açısından bakıldığında da durum aynıdır; Genel bir DF tasarımı çok zor ve uzun zaman isteyen bir iştir. Çünkü çok farklı amaçlara hizmet eden CBS ler arasında veri değişiminin olanaklı olabilmesi için, DF nin farklı CBS veri modellerini tanıması, diğer bir ifadeyle bu modelleri kapsamaları gerekir. Kaldı ki böyle bir modelin tasarımında, gelecekteki gereksinimleri kestirmek çok zor olduğundan, DF nin zamanla genişletilmesi kaçınılmazdır [14], [21], [81]. O nedenle DF nin veri modelinin "genişletilebilir" olması çok büyük bir önem taşımaktadır.

Bir veri modelinin yüksek derecede bir genişletilebilirlik sunabilmesi, genelleştirme/özelleştirme, birleştirme/ayırıştırma, ilişkilendirme gibi veri soyutlama mekanizmalarını desteklemesiyle mümkün olacaktır. Çünkü bu mekanizmalar yardımıyla genişletme için gereksinim duyulan yeni veri tipleri mevcut veri tiplerinden yararlanarak tanımlanabilir. Bu bakımdan ND model alışıl gelmiş kayıt-tabanlı modellere çok önemli üstünlükler sağlamaktadır. Bu konuda daha ayrıntılı bilgi için [100]'e başvurulabilir.

DF nin genişletilebilir bir veri modeline sahip olmasının interoperabilite ve dolayısıyla UKVA açısından önemi çok büyüktür. Konumsal veri yönetiminde bugün artık kaçınılmaz hale gelen interoperabilite, yazılım ve/veya donanım olarak farklı CBS lerin federe bir yapı içerisinde, birbirleriyle ortaklaşa (bütünleşik) bir tarzda çalışabilmelerini ve aynı veriyi "paylaşabilmelerini" gerektirir. Diğer bir anlatımla interoperabilite ile kastedilen, federe bir sistemde bir uygulama programının gereksinim duyduğu, değişik veri tabanlarında bulunan veriye, programcının müdahalesi olmaksızın erişebilmesidir [119]. Bunun için, farklı sistemlerin farklı veri gösterimleri arasında geçişi sağlayacak, ortak bir veri modeline (common/canonical data model) gereksinim vardır [15], [120]. Bu veri modeli genel, esnek, ve aynı zamanda da genişletilebilir olmalıdır. İnteroperabilite, UKVA açısından son derece önemlidir. Çünkü UKVA ile en yüksek yarar, UKVA içerisinde federe sistemlerin gerçekleştirilebilmesi ile mümkün olacaktır. Dolayısıyla, DF nin veri modelinin sözkonusu gereksinimi karşılayabilmesi son derece önemlidir.

2.2 Gerçekleştirim Ölçütleri

VTYS ler bağlamında "gerçekleştirim" dendiğinde, kullanılan veri yapıları, işletim sistemi ve donanım bağımsızlığı, iletişim protokolları ile etkileşim gibi konular akla gelir [40]. Veri modeli ve gerçekleştirim arasında, bazen göz ardı edilen, ayırım önemlidir. Çünkü verilen bir model çok değişik şekillerde gerçekleştirilebilir. Örneğin ND bir arabirim ile kullanıcılarına gereksinim duydukları nesneleri oldukça soyut bir düzeyde sunan bir CBS, ilişkisel modele dayalı olarak gerçekleştirilebilir. Böylece kullanıcı yalnızca ilgi alanına giren nesneleri görür ve onları kullanarak problemlerini modeller, ancak işlemler CBS nin kullandığı İVTYS de gerçekleştirilir. Böylece gereksiz gerçekleştirim detayları kullanıcılardan gizlenmiş olur.

DF ler bağlamında gerçekleştirim, formatın bilgisayar ortamında bir veri değişiminde geçerli olan, formatın özellikli transferler için uyarlanabilmesi, verinin fiziksel olarak kodlanması, detay kodlama vs. gibi ayrıntılar anlaşılır. Çoğu alışıl gelmiş DF dökümanında, veri modeli ve gerçekleştirim kavramları arasındaki ayırımın yeterince açık olmadığı bir

gerçektir [76]. Sözkonusu ayırım genelde, donanımdan bağımsız ("taşınabilir") yazılım üretimi ve dolayısıyla standartlaştırma açısından son derece anlamlıdır. Bu açıdan en önemli kilometre taşlarından biri, ISO (International Standards Organization)'nun OSI (Open Systems Interconnection) modeli (bkz. 2.2.3) dir. Veri modeli ve gerçekleştirimin birbirinden bağımsız olarak ele alınması, DF nin donanım ve iletişim ortamı bağımsızlığının sağlanması için zorunludur. Bu nedenle sözkonusu ayırım, yeni DF lerde alışlagelmiş DF lere göre çok daha belirgindir.

Gerçekleştirim boyutunun DF tanımının dışında tutulması, yeni geliştirilen bir DF nin standartlaşmasını olumsuz yönde etkileyebilir. Genel ve esnek bir DF nin oldukça karmaşık olacağı, dolayısıyla gerçekleştiriminin de ayrı bir tasarım ve uzmanlık gerektirdiği kabul edilmektedir [40], [121]. Dolayısıyla, çevirici geliştirme güç ve zaman alıcı bir iştir. Bu yükten kaçınmak ve zamandan kazanmak isteyen kullanıcılar yeni format için tamamen yeni bir çevirici geliştirmek yerine, kullanmakta oldukları formatlarla yetinmeyi yeğleyebilir. Ya da farklı kullanıcılar, formatı farklı şekillerde uygulayabilir. Bunun sonucunda da, standartlaştırma ile amaçlanan tamamen tersine olarak, aynı format için farklı çeviriciler ortaya çıkabilir [83], [122]. Bu nedenle DF nin veri modeli tanımının yanında, gerçekleştirim detaylarını da kapsamı, çeviricilerin kolay ve hızlı bir şekilde geliştirilmesine ve etkin bir veri değişimine olanak tanıyacak servis ve araçlar ya da bunları içeren bir Grafik Kullanıcı Arabirimi (GKA) yı kapsamında bulundurması son derece önemlidir. Örneğin, farklı formatlar için kodlayıcı ve kod çözücü yazılımlarının DF ile birlikte gelmesi, çevirici geliştiricilere önemli kolaylıklar sağlayacak ve formatın standartlaşmasını çok önemli ölçüde teşvik edecektir.

DF nin gerçekleştirimi ile ilgili üç temel ölçüt belirlenmiştir. Bunlar DF yi, formatın özellikli transferler için uyarlanabilirliği, nesne sınıflaması ve detay kodlama stratejisi ile, kullanılan fiziksel kodlama yöntemleri açısından değerlendiren ölçütlerdir. Aşağıda, bu ölçütler tanıtılmaktadır.

2.2.1 Profiller

Genel olduğu kadar esnek bir DF, oldukça karmaşık olacağı için, formatın genel bir çevirici ile gerçekleştirimi, pek çok açıdan pratik bir çözüm değildir. Öncelikle, genel bir çeviricinin geliştirilmesi uzun zaman alabilir ve pahalı olabilir [84]. İkinci olarak, genel bir çevirici belli bir veri değişiminin genel amaçlı bir DF ye göre çok özellikli olabilmesi nedeniyle gereksizdir. O nedenle, ya özellikli bir DF kullanılmalı ya da genel amaçlı DF, bir şekilde amaca özel bir hale getirilmelidir. Örneğin vektör veri kullanan bir CBS nin raster modunda veri ithal etmesi anlamsızdır. Benzer şekilde, vektör veri kullanan bir CBS nin,

veri transferlerinde hem raster ve hem de vektör veri transferi için geliştirilmiş bir DF kullanması fazladan yük getirecektir. Nihayet genel bir çevirici, esneklik için gerekli tüm seçenekleri dikkate almak durumunda olduğundan verimsiz olacaktır. Örneğin transfer veri grubunun yalnızca otoyollardan oluştuğu bir durumda, genel bir çeviricinin koşum anında diğer karayolu tiplerini de dikkate almak durumunda olması verimi olumsuz etkileyecektir. O nedenle genel bir DF nin, amaca özel alt-grupları kullanılarak gerçekleştirilmesi çok daha pratiktir.

Bir DF nin amaca özel alt-grupları "profil" olarak anılır [22], [84] (bkz. 1.3). Raster ve vektör veri transferine izin veren bir DF nin, yalnızca vektör veriye yönelik bir profili gibi. Diğer bir örnek hem metrik hem de topolojik veriyle başa çıkabilen bir DF nin, yalnızca metrik veriye yönelik profili olabilir. Profil belirtme, profilin izin verdiği veri tipleri ile, bu veri tiplerinin alabileceği değerler ve çeşitli kısıtlayıcıların belirlenmesini içerir. Bunun için karşılaşılabilecek veri değişimleri dikkate alınmalıdır. Örneğin vektör ve raster veri için iki ayrı profil tanımlanması uygun olacaktır.

Profiller kullanıldığında veri değişiminde bulunan taraflar arasında profiller üzerinde anlaşma sağlanmalıdır. Bu bakımdan profillerin tanımlanması ve yaşatılmasının belli bir otorite tarafından yapılması gerekir. Böylelikle herhangi bir profile göre tanımlanmış veri her iki tarafa da aynı anlam ifade edecektir. Profiller yakın zamanda geliştirilen ve profilli gerçekleştirimi destekleyen NTF [85], SDTS [22], ve SAIF [35] gibi formatlarla, alışlagelmiş formatların uyumluluğunun sağlanması açısından da son derece kullanışlıdır. Çünkü DXF, SIF, DLG gibi alışlagelmiş formatların kapsamı yeni formatlara göre çok daha dardır.

2.2.2 Detay Kodlama

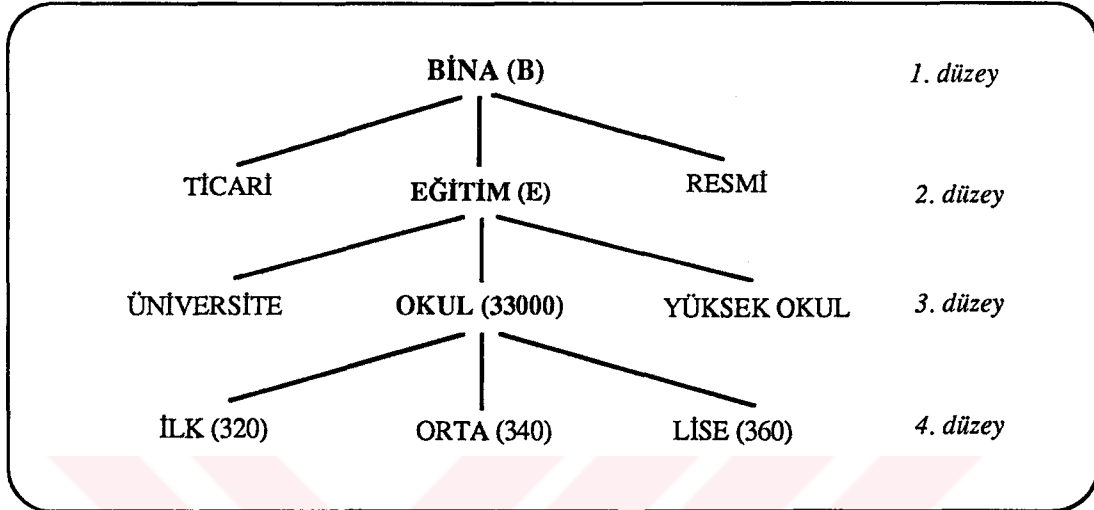
Coğrafi bir veri tabanı havaalanı kulesi, su deposu, enerji santrali, karayolu, demiryolu, akarsu, göl, parsel gibi çok çeşitli coğrafi nesnelerden oluşur. "Coğrafi nesne", yer üstü ve yer altında bulunan ve konumsal bir tanımı olan varlık, olarak tanımlanmaktadır [40]. Konumsal tanım metrik ya da topolojik olabilir [39]. Her coğrafi nesnenin kendine has bir takım özellikleri yani öznitelikleri vardır. Örneğin havaalanı kulesi ve su deposunun yüksekliği ve genişliği, karayolunun uzunluğu ve şerit sayısı, akarsuyun uzunluğu ve debisi, gölün alanı, parselin alanı, sahibi, değeri gibi öznitelikleri sözkonusudur. Teorik anlamda, coğrafi nesnenin konumu da tanımlayıcı bir özellik, dolayısıyla bir özniteliktir. Bu nedenle, yazara göre "tanımsal-konumsal" ayrımı yanlıştır. Bu bakımdan öznitelikler konumsal ve konumsal-olmayan öznitelikler olarak iki grupta düşünülebilir.

"Coğrafi nesne" terimi nesne-tabanlı veri modelleri ile birlikte düşünülür. Bugün kullanımda olan çoğu CBS ise, daha önce de belirtildiği gibi kayıt-tabanlıdır. Bu sistemler bağlamında coğrafi nesneye karşılık gelen sözcük "detay" dır. Bu sistemlerde konumsal bilgi, nokta, çizgi, ve poligon gibi temel geometrik elemanlar yardımıyla gösterim edilir. Diğer bir anlatımla nokta, çizgi, ve poligon gibi temel geometrik elemanlar, havaalanı kulesi, nükleer santral, su deposu, karayolu, nehir, göl, parsel gibi varlıkları gösterim ederler (bkz. 1.3). Bu geometrik elemanların gerçekte hangi tip varlıkları gösterim ettiklerini belirtmek üzere detay kodları kullanılır. "Detay kodu" (DK) bir varlık sınıfını tanımlar, tek bir varlığı tanımlamaz. Belirli bir varlığı tek anlamlı olarak tanımlamak için anahtarlar kullanılır. DK lar belli bir uzunlukta, "AL015" gibi karakter ya da "0225045" gibi sayısal olarak ifade edilebilir.

Detay kodları, varlıkların karakteristiklerine göre belirlenir. Örneğin 2 şeritli beton asfalt karayolları için "KYBA2Ş" kodu kullanılabilir. Detay kodlama, varlıkların karakteristiklerine göre sınıflandırılmasını ve her bir sınıfa bir DK nın atanmasını içerir. Bu işlemin iki temel aşaması, kodlama tekniğinin ve kodlamanın ayrıntı düzeyinin belirlenmesidir. Hiyerarşik kodlama, alışlagelmiş bir kodlama tekniğidir. Bu teknikte sınıflandırma veri soyutlama mekanizmaları olarak bilinen genelleştirme ve özelleştirme ye dayanmaktadır. Buna göre örneğin, Şekil 27 de ORTA OKUL, OKUL un özel bir halidir ve DK sı 33000340 tır. Hiyerarşik detay kodlama ile yüksek bir veri erişim hızı sağlanabilir. Örneğin yalnızca binalar için "B" ile, yalnız okullar için de "BE33000" ile başlayan detaylar aranacaktır. Bununla birlikte hiyerarşik kodlama, aynı tipte detaylar için farklı DK lar elde edilebilmesi, yani öznel bir yöntem olması nedeniyle eleştirilmektedir [11]. Karayolu örneğinde, hiyerarşide yolun beton asfalt olması, çift şeritli olmasından daha alt düzeyde düşünülürse, yani beton-asfalt-karayolu çift-şerit-karayolunun alt ya da daha özel bir sınıfı olarak alınırsa, DK yukarıdakinden farklı olarak, "KY2ŞBA" şeklinde elde edilecektir. Hiyerarşik olmayan kodlamada DK lar belli bir hiyerarşiye uymadıklarından böyle bir problem söz konusu değildir.

Kodlamanın ayrıntı düzeyi "kodlama çözünürlüğü" olarak tanımlanabilir. Bu, hangi detay özniteliklerinin kodlamada dikkate alınacağı ile ilişkilidir. Bütün özniteliklerin alması olası her değer dikkate alınarak, bu değerlerin her farklı kombinasyonu için farklı bir DK kullanılması, kodlama çözünürlüğünü artırırken, kodlama işlemini güçleştirecek ve gerçekleştirimi hantallaştıracaktır. O nedenle kodlamanın sınırlı sayıda öznitelik üzerinden yapılması daha pratik bir çözüm olacaktır. Bütün sorun bu özniteliklerin belirlenmesidir. Bunun için en önemli ölçüt, özneliğin sınıflandırmada dikkate alınmasının, konumsal veri kullanıcı kesimleri arasında ortak bir kabul görmüş olmasıdır. Buna ek olarak, kodlama çözünürlüğünün artırılmasında özneliğin alması olası değerler (domain) dikkate alınmalıdır;

Şu kadar ki, domain genişliği ile bunun kodlamaya getireceği ek yük doğru orantılıdır. DK karakter uzunluğu ile DK'nın karakter ya da sayısal olarak ifade edilmesi, kodlama çözünürlüğü ile doğrudan ilişkilidir. Belli bir uzunlukta karakter modunun sağlayacağı çeşitlilik daha fazladır.



Şekil 27. Hiyerarşik Detay kodlama.

ND modelleme, varlıkların özniteliklerine göre sınıflandırılmasına dayanmaktadır. ND modelde sınıflandırma hiyerarşisinin iki boyutu vardır (bkz. 1.2). Bu hiyerarşi içerisinde her sınıf bir dizi öznitelik içermektedir. Örneğin bir karayolu sınıfı kaplama, şerit sayısı, ve mevcut-durum gibi özniteliklerle tanımlıdır. Bu özniteliklerin domainleri bellidir. Dolayısıyla, ND modelde alışlagelmiş şekliyle detay kodlamaya gerek yoktur. Gerçekleştirmede, sözkonusu domain değerlerinin her biri için bir kod gerekecektir tabi, ancak alışlagelmiş kodlamaya kıyasla burada kodlar otomatik olarak türetilir. Böylelikle hem tek tek kimliklendirilmesi ve tanımlanması gerekli detay sayısı azalmış olur, hem de ayrıntılı DK listeleri oluşturma ve bunları ilgili tanımlamalarla ilişkilendirme yükü, kullanıcıların sırtından alınmış olur. Bunu bir örnekle açıklamak gerekirse, yukarıdaki karayolu örneğinde her üç öznitelik domaininin de üç değerle sınırlı olduğu kabul edilsin. Bu üç değere ek olarak bir de "boş" değeri olabilir. Bu durumda alışlagelmiş detay kodlamada, 64 farklı detayın kimliklendirilmesi ve tanımlanması gerekir. Buna karşılık ND bir gerçekleştirmede, üç öznitelik ve dokuz değer ve bir de boş olmak üzere yalnızca 13 kimliklendirme ve tanımlama gereklidir.

Detay kodlama, başlıbaşına bir standartlaştırma alanı durumundadır [12]. Bu amaçla çeşitli detay kodlama standartları geliştirilmiştir. FACC (Feature and Attribute Coding Catalogue) [123]. bunlardan biridir. Bu bakımdan bir DF kendine özel ("iç") bir detay sınıflamasını kapsayacağı gibi, FACC gibi bir "dış" standart da kullanabilir. Ya da her iki

seçeneği de birlikte sunabilir ki bu en çok arzu edilenidir. Ayrıca DF, iç ve dış kodlama katalogları dışında kalan özniteliklerin (ad-hoc öznitelikler) bir "veri sözlüğü" yardımıyla tanımlanarak transferine izin vermelidir.

2.2.3 Kodlama Yöntemi

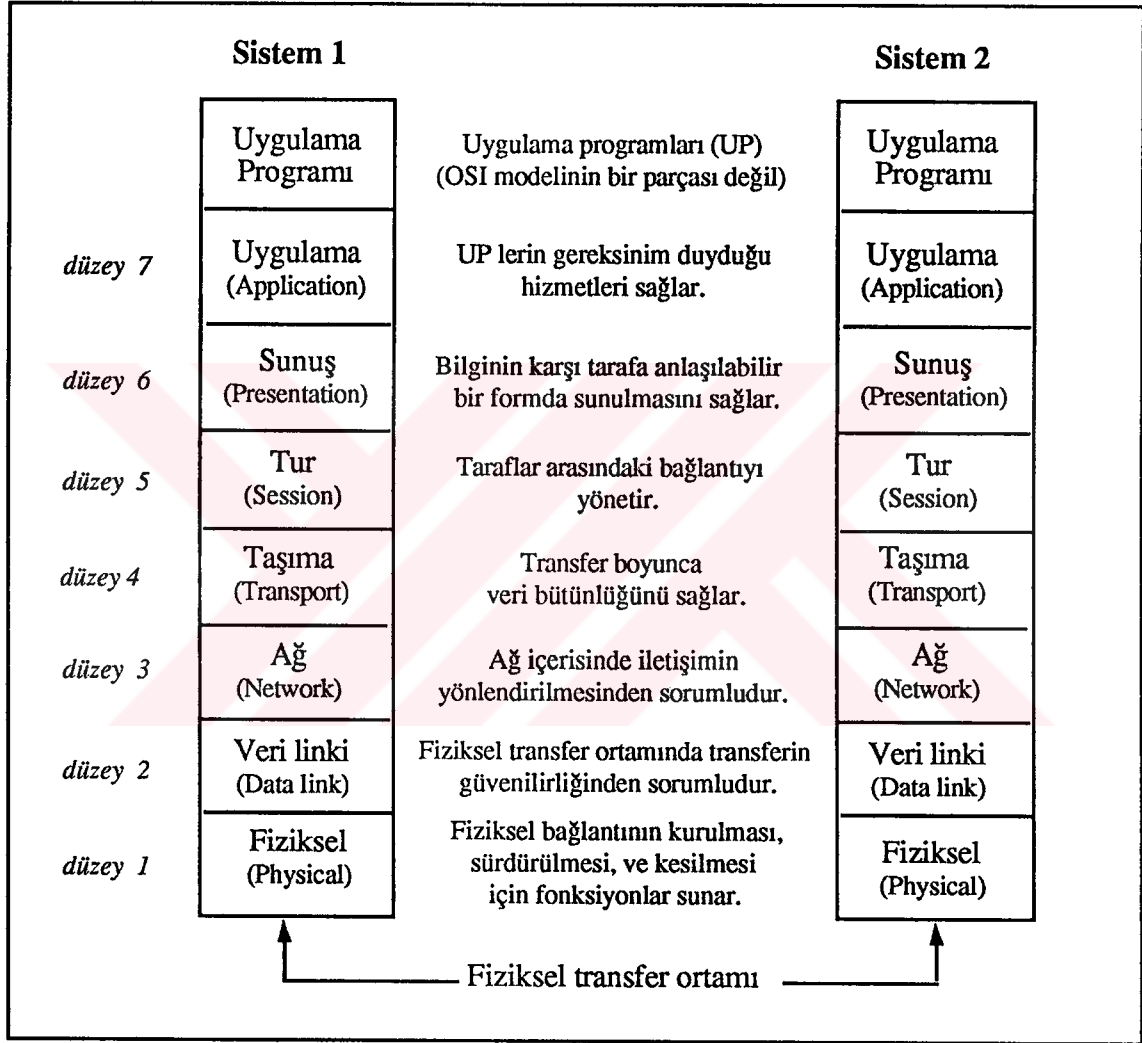
Veri değişiminde verinin sunucudan alıcıya fiziksel transferi çevrim-içi ya da çevrim dışı bir tarzda gerçekleştirilebilir. Her iki durumda da, belirli bir veri yapısındaki verinin, bit/Byte lardan oluşan bir veri dizisi olarak kodlanması gerekir. Kodlama işlemi, sayısal veri dizisinin yorumlanabilmesi için bir takım kural ve kodlarla anlamlandırılmasını içerir. Bunun için çeşitli kodlama teknikleri ve standartları mevcuttur (bkz. 1.3).

Bilgisayar sistemleri, kayıt ortamları, ve iletişim hatlarının çeşitliliği DF tasarımında DF nin soyut tanımı ile kodlama yönteminin birbirinden ayrılmasını zorunlu kılmıştır. Buradaki ayırım, 1.2 de sözedilen kavramsal ve iç şemalar arasındaki ayırma benzerdir ve DF lerin donanımdan bağımsız olarak tanımlanması için gereklidir. Buradaki temel ilke, standartlaştırmanın farklı düzeylerde sağlanmasının çok karmaşık bir işlem olan standartlaştırmayı daha basite indirgeyeceğidir. Bu aynı zamanda, ISO (International Standards Organization)'nun OSI (Open Systems Interconnection) modelinin [71] de temelini oluşturmuştur. OSI modeli iletişim ağlarının standartlaştırılması için yedi ayrı düzey tanımlamaktadır. Şekil 28 de görüldüğü üzere en üstte uygulama düzeyi, en altta da fiziksel düzey yer alır. Her bir düzey alt ve üst düzeyleri ile belli standartlara (protokol) göre iletişim sağlar. Böylece sistem 1 den sistem 2 ye doğru bir iletişim, sistem 1 de ilgili Uygulama programı (UP) tarafından 7. düzeyden başlatılır. Daha sonra ilgili ileti ya da veri 7. düzeyden 1. düzeye kadar bütün düzeylerden geçerek, her düzeyde o düzeyin gerektirdiği şekilde ve o düzey için geçerli standartlara göre düzenlenerek fiziksel transfer ortamı üzerinden sistem 2 ye gönderilir. Sistem 2 de 1. düzeyde alınan veri, sistem 1 dekinin tersi bir işlemten geçerek ilgili UP ye ulaşır.

ISO-OSI modelinin her düzeyi için kabul edilmiş standartlar mevcuttur [69]. OSI modeline göre iletişimin ayrıntıları bu tezin kapsamı dışındadır. Şu kadarını belirtmek gerekirse, farklı organizasyonlar (ISO, IEEE, CCIT) tarafından geliştirilmiş olan bu standartlar, tam olarak tek bir düzeye karşılık gelmeyip, farklı düzey standartları arasında bindirmeler olabilmektedir. Örneğin TCP/IP (Transfer Control Protocol/Internet Protocol) 3. ve 4. düzeylere, Ethernet (IEEE 802.3) 1. ve 2. düzeylere karşılık gelmektedir.

Veri değişiminde transfer veri grubunun kodlanması, OSI modelinin sunuş düzeyi kapsamındadır. Verinin belli bir formda sunulması ile ilgili olarak ISO, "soyut" ve "somut

sözdizim" kavramlarını getirmiştir. Belirli bir soyut sözdizimine uyan transfer veri grubunun fiziksel olarak kodlanması, somut sözdizime göre yapılır. Soyut ve somut sözdizim için çeşitli standartlar geliştirilmiştir. Soyut sözdizim için en çok, BNF (Backus-Naur Form) [77] ve ASN.1 (Abstract Syntax Notation) (ISO 8824) [74], somut sözdizim için de, ISO 8211 [78] ve ASN.1 kodlama kuralları [74] dır. Bu standartlar ve kodlama hakkında ayrıntılı bilgi, bölüm 3 te verildiği için burada ayrıca değinilmeyecektir.



Şekil 28. ISO-OSI modeli.

Transfer veri grubunun kodlanmasında, etiketli ya da etiketsiz kodlama uygulanabilir. Ayrıca veri çoğu DF lerde olduğu gibi ASCII ya da ikili modda kodlanabilir. Bu seçeneklerin avantaj/dezavantajları vardır. Örneğin ASCII kodlama, ikili modun aksine donanımdan bağımsız olmasına karşılık, ikili moda göre daha fazla kayıt alanı gerektirir. Benzer şekilde, etiketli kodlama daha fazla kayıt alanı gerektirmesine karşılık, etiketsiz kodlamadaki gibi alıcının transfer veri grubunun soyut sözdizimini bilmesini gerektirmez. Somut sözdizim standartları değişik seçenekler sunmaktadır. Örneğin ISO 8211 de, ASCII

veya ikili kodlama yapılabilir ve etiketli kodlama uygulanır. ASN.1 de etiketli kodlama uygulanmakla birlikte, özlü kodlama desteklenmektedir (bkz. 1.3).

DF nin kullandığı kodlama yönteminin kısa ve özlü bir kodlamayı desteklemesi gerekir. Bu, özellikle çevrim-içi veri transferinde, transferin hızı bakımından, daha fazla önem kazanmaktadır. Ayrıca kodlama iletişim ortamından bağımsız olmalıdır. Bu bakımdan DF'nin farklı somut sözdizim standartları ile birlikte kullanılabilmesi uygun olacaktır. Örnek olarak, DF nin çevrim-dışı transferlerde ISO 8211 i, çevrim-içi transferlerde ASN.1 kodlama kurallarını kullanabilmesi verilebilir.

2.3 Değişim Ortamı

Konumsal veri değişiminde sözkonusu olan, bir tarafın diğer taraftan veri istemesi, buna göre gerekli verinin derlenmesi ve bir şekilde transfer edilmesidir. İki tarafın aynı veriyi gösterim farklılıkları bir yana, burada en önemli sorun, taraflar arasında iyi bir iletişim kurulmasıdır. Bu iletişim sayesinde, daha işin en başında alıcı, sunucunun gerekli veriyi sağlayıp sağlayamayacağını ya da ne oranda sağlayacağını belirleyecek ve "neyi", "nasıl" istediğini belirtecektir. Bu iletişim her iki tarafın anlayabileceği ortak bir dil kullanmalıdır. Sözkonusu iletişim için konumsal veri değişim ortamının kullanıcılar için gerekli araç ya da arabirimleri sağlaması gerekir.

"Değişim ortamı" ile, DF nin bilgisayar ortamında gerçekleştiriminde gerek çevirici geliştirme ve gerekse veri transferini yerine getirmede gerçekleştirmecilere/kullanıcılara sunulan destekleyici yazılımlara (arabirimler/araçlar) işaret edilmektedir. Konumsal veri değişimi alanındaki araştırmalar bugüne kadar hep DF tasarımı üzerinde yoğunlaşmış, bu formatların nasıl gerçekleştirileceği ya da uygulanacağı konusu çoğunlukla gözardı edilmiştir. Bunun sonucu olarak konumsal veri değişiminin bugün pratikte uygulanışına bakıldığında, veri gruplarını görme, sorgulama ve ardından istenen alt grupları ya da nesneleri transfer etmeye olanak tanıyan Grafik Kullanıcı Arabirimlerinin (GKA) eksikliği göze batmaktadır [15], [40].

Değişim ortamı ile ilgili olarak üç ölçüt belirlenmiştir. Bunlar formatın dokümantasyonu, çevirici geliştirme desteği ve değişim ortamı araçları ile ilgili ölçütlerdir. İzleyen kısımda bu ölçütler tanıtılmaktadır.

2.3.1 Dokümantasyon

Genel ve aynı zamanda esnek bir DF nin oldukça karmaşık olacağı ve gerçekleştiriminin de ayrı bir tasarım ve uzmanlık gerektirdiği kabul edilmektedir [40],

[121]. [40], bir SAIF gerekleřtiriminde eřitli kullanıcıların bilmesi gereken detayların bir listesini sunmaktadır. DF lerin dokümantasyon yetersizlięi ve ortalama kullanıcı tarafından anlaşılma gúçlüęü sıka dile getirilen bir problemdir [124]. Oysa iyi organize edilmiř, açık ve kolaylıkla anlaşılabilir bir dokümantasyon, gerekleřtirmcilerin iřini kolaylařtıracak ve DF nin standartlařmasını önemli ölçüde teřvik edecektir. Bunun için öncelikle, gerekleřtirim veri modeli kavramları arasındaki ayırımın, oęu alışıl原因lmıř DF dökümanında görüldüęünün tersine, net bir biçimde yapılmıř olması gerekir. Bunun yanında, DF dökümanının iyi bir řekilde düzenlenmiř olması, yaygın olarak kabul edilmiř bir terminoloji kullanması ve1 kavram kargařasına yer bırakmaması gerekir.

2.3.2 Servisler

"Servisler" ile, evirici, kodlayıcı, kod özücü, ve iletiřim rutinleri/modúlleri kastedilmektedir. Bu servisleri ieren bir deęiřim ortamı ile, eviriciler ya da "deęiřim arabirimleri" hızlı ve kolay bir biçimde, ya da oęunlukla anılan řekliyle "modúller" tarzda, geliřtirilebilecek ve arabirimler yine bu servisler vasıtasıyla iřlevlerini yerine getirecektir. Örneęin, alışıl原因lmıř CBS lerin topolojik veri yapıları önemli benzerlikler tařır. Bu veri yapıları arasında geiři saęlayan eviri rutinleri, bu veri yapılarına dayalı veri gruplarını ieren her veri deęiřiminde kullanılabilir. evirici rutinler, 1.3 te söz edilen, bir sayısal veri gösteriminden dięerine geiři için gerekli evirileri gerekleřtiren rutinlerdir. Kodlayıcı ve kod özücü rutinler ise verinin, belirli bir kodlama yöntemi ya da standardına göre kodlanması ve kod özümleme iřlemlerini yerine getiren rutinlerdir. İletiřim servisleri ise, verinin bir taraftan dięerine fiziksel olarak aktarımı için gereklidir. İletiřim servislerine örnek olarak, dosya transfer, masaj gönderme, ve diyalog servisleri verilebilir [29]. DF nin böyle bir deęiřim ortamı ile desteklenmesi, formatın yaygın kabul görmesinde ok önemli bir rol oynayacaktır. Burada bütün sorun, servislerin nasıl organize edileceęidir. Bu konu, ařaęıda ele alınmaktadır.

eviriler, basit ya da karmařık olabilir. Uzunlukların bir tarafta *inch*, dięerinde *cm* olarak ifade edilmesi basit eviriler gerektirir. Karmařık eviriler ise, birleřtirme/ayrıřtırma ve topoloji oluřturma gibi iřlemler ierir. Karmařık eviriler, katman-tabanlı ve nesne-tabanlı modeller arasında söz konusudur (bkz. 1.3). Örneęin bir eřyükseklik eęrisi ve onun yükseklik deęeri katman-tabanlı tarafta İZGİ ve İSİM gibi, farklı veri tipleri olarak iřlem görebilir. Nesne-tabanlı tarafta ise, yükseklik deęeri de YÜKSEKLİK_EĞRİSİ nesnesinin bir elemanı olarak görülebilir ve yükseklik eęrisine uygulanan iřlemler bu nesnenin yükseklik deęeri ve dięer bileřenlerini de etkileyebilir. örneęin eęrinin silinmesi ile yükseklik deęeri de silinmiř olur. Aynı sonuç, katman-tabanlı tarafta iki ayrı silme iřlemi—

ÇİZGİ ve İSİM silme—gerektirir. Bu durumda nesne-tabanlı tarafa geçişte, ilgili çizgi ve isimin birleştirilerek yükseklik_eğrisinin oluşturulması gerekecektir.

Modülerlik ve tekrarlı kod kullanımı için alışlagelmiş yaklaşım, "rutin kitaplıkları" kullanmaktır. Ancak bu yöntem, sınırlı derecede bir tekrarlı kod kullanımını olanaklı kılar. Çünkü alışlagelmiş programlama yaklaşımı ile, rutinlerin gerek kullandıkları veri yapıları ve gerekse işlevleri bakımından, birbirlerinden keskin sınırlarla ayrılması oldukça güçtür. Bindirmelerin önlenmesi için rutinlerin işlevleri bakımından küçültülmesi ise, rutinlerin organizasyonunu ve tekrarlı kod kullanımını güçleştirecektir [64]. Çünkü, rutin kitaplıklarının organizasyonu oldukça karmaşık bir iştir. Bu durum, zamanla bu kitaplıkları farklı kullanıcıların kullanması ve bunların da yeni rutinler eklemesi ya da mevcut rutinler üzerinde değişiklikler yapmaları sonucu daha da kötüleşecektir.

Alternatif olarak, çıkış noktası modüler ve genişletilebilir yazılım üretimi olan NY tekniği kullanılabilir. ND veri modelinde nesneler öznitelikleri yanında, kendilerine uygulanabilecek işlemleri—rutinler—de tanımlarlar (bkz. 1.2). Dolayısıyla, farklı topolojik veri yapıları arasında çeviriler ile birleştirme/ayırıştırma içeren çeviriler de dahil tüm çeviri rutinleri, kodlayıcı ve kod çözücü rutinler DF nin ilgili nesne sınıfları altında tanımlanabilir. İletişim rutinleri de, değişim ortamının ilgili nesne sınıfları içerisinde tanımlanabilir. Gerçekleştirimde, çevirici bu rutinleri ilgili nesneler üzerine uygulayarak dönüşümü sağlayabilir. Böyle bir değişim ortamının diğer avantajı, rutinlerin işlevsel sınırlarının nesneler bazında belirlenebilmesi ve rutinlerin ilgili nesnelere bağlanmasıdır. Böylelikle, alışlagelmiş programlama ortamlarında oldukça karmaşık bir iş olan rutin kitaplıklarının organizasyonu, çok daha kolay bir hale gelir. ND veri değişim ortamı hakkında ayrıntılı bilgi 3. bölümde verilmiştir.

Servisler, ND yaklaşım açısından bakıldığında, veri modeli ile de çok yakından ilişkilidir. Ancak bu ölçüt ile aranan, yukarıda anılan servisleri içeren bir değişim ortamı desteğidir. Veri modeli desteği ancak, DF nin ND bir veri modeline sahip olması durumunda söz konusudur. Yukarıda açıklandığı gibi ND bir veri modeli ve ND bir gerçekleştirim servisler açısından idealdir. Ancak alışlagelmiş veri modeline dayalı bir DF için de servis desteği gerçekleştirimde, değişim ortamı ile verilebilir. O nedenle bu ölçüt, bir veri modeli ölçütü olarak değil de, değişim ortamı ölçütü olarak alınmıştır. Dolayısıyla, servisler ve veri modeli açısından üstünlük, "veri modeli tipi" ölçütü ile ima edilmektedir.

2.3.3 Değişim Ortamı Araçları

Derleyici/Yorumlayıcı araçları : Çevirici, kodlayıcı, ve kod çözücü rutinlerin DF nin soyut ve somut tanımlarıyla birlikte gelmesi, DF için çevirici geliştirmeyi kolaylaştıracak ve DF

nin yaygın kullanım bulması açısından çok önemli bir rol oynayacaktır. Bununla birlikte, konumsal veri değişimi "uygulama bağımlı"dır. Diğer bir ifadeyle, farklı uygulamalar çok farklı tipte verilere gereksinim duyabilirler. Bu nedenle, iki farklı format arasında özellikli amaçlara yönelik çeviriciler geliştirmek ya da mevcut genel bir çeviriyi bu amaçla uyarlamak, çevirici, kodlayıcı, ve kod çözücü rutinler için bir rutin kitaplığı bulunsa bile, çevirici geliştiriciler için güç ve zaman alıcı bir iştir. Bu nedenle çevirici geliştirmede son yıllarda gözlenen eğilim, derleyici ve yorumlayıcı teknolojilerinden yararlanılarak çeviricilerin otomatik olarak geliştirilmesidir. Böylelikle anılan yük programcının sırtından alınacak ve çeviriciler çok daha kısa bir zamanda geliştirilebilecektir (bkz. 1.3).

Derleyici/yorumlayıcı teknolojisinin konumsal veri değişiminde kullanılabileceğinin çıkış noktası konumsal verinin iki farklı gösterimi arasındaki çevirinin, programlama dili ile makine dili arasındaki çeviriye benzerliği olmuştur. Buna göre, verinin transferden önce, transfer sırasında, ve transferden sonraki gösterimleri ile çeşitli veri gösterimleri arasındaki çeviriler belli bir sözdizime göre resmi olarak tanımlanabilirse, çevirici program kodu otomatik olarak üretilebilir ya da yorumlayıcı modunda çeviri gerçekleştirilebilir (bkz. 1.2). Burada verinin transferden önceki sonraki gösterimlerinden kasıt, verinin konumsal veri değişiminde bulunan taraflardaki gösterimleridir. Transfer sırasındaki gösterim ise, DF nin soyut ve somut tanımına uyan gösterimdir. Veri değişiminde iki taraf arasında bir DF nin kullanılmaması, yani doğrudan çeviri durumunda bu tanımlamaya gerek olmayacaktır. Sonuç olarak bir DF nin derleyici araçları—tarayıcı, ayrıştırıcı—içeren bir değişim ortamı ile desteklenmesi, formatın kullanımını çok önemli ölçüde teşvik edecektir.

Veri Tanımlama Dili (VTD) : DF nin soyut tanımının, farklı kullanıcılar tarafından yanlış yorumlamalara yol açmaması için, bir RDT de yapılmış olması gerekir. RTD, metinsel ya da grafik olabilir. DF nin metinsel ve grafik bir VTD sunması, kullanıcıların soyut tanımı daha iyi anlamaları açısından son derece yararlı olacaktır. Otomatik çevirici geliştirmede, çeviricinin gereksinim duyduğu tanımlamaların da metinsel bir VTD de yapılması gerekir. İdeal olarak, metinsel ve grafik VTD de arasındaki çeviri, otomatik olarak gerçekleştirilebilmelidir [29]. Ayrıca DF nin soyut tanımının, çeşitli ayrıntı düzeylerinde çevrim-içi incelenmesini kolaylaştıracak, yardımcı yazılımlar (Browsing software) da yararlı olacaktır. ND sistemlerde sınıf hiyerarşilerini grafik olarak gösteren yazılımlar bu tür yazılımlara bir örnektir. Böyle bir yazılım, verilen bir sınıf için alt ve üst sınıfları, sınıf ve nesne öznitelik ve servislerini, sınıfın sınıf oluşum hiyerarşisi içerisinde ilişkide olduğu diğer sınıfları vs. gösterir [125].

VTD, DF nin değişen gereksinimlere göre genişletilmesi için de gereklidir. Çünkü kullanıcıların sözkonusu genişletmeleri de VTD de tanımlamaları yine hemen yukarıda anılan nedelerle ve DF nin sürekliliğinin sağlanması açısından gereklidir.

Seçmeli Transfer Araçları : İdeal bir veri değişim ortamında alıcı, sunucu verisini bir GKA vasıtasıyla görüp, sorgulayarak yalnızca gereksinim duyduğu veriyi transfer edebilmelidir. Diğer bir ifadeyle, "seçmeli transfer" ya da "nesne bazında transfer" (bkz. 2.1.9) olanaklı olmalıdır. Alıcının gereksinim duymadığı veriyi transfer etmesi, bu veriye kodlama, transfer, kod çözümü, ve olası "gereksiz verinin ayıklanması" işlemlerinin uygulanmasını gerektirecek, transfer zamanını artıracak, sonuçta kaynak israfına neden olacaktır. Modern CBS lerin buna tahammülü yoktur. Konumsal veri değişiminin bugün pratikte uygulanışı ele alındığında, seçmeli transfer en önemli eksikliklerden biri olarak ortaya çıkmaktadır. Çünkü henüz, seçmeli transfer araçlarına sahip bir GKA destekli bir DF bulunmamaktadır. DCW (Digital Chart of the World) ürünü, kullanıcılarına veriyi görme, tarama ve istenen veriyi transfer etme olanağı tanımaktadır. Ancak DCW bir DF değil, 1:1 000 000 ölçeğinde dünya genelinde çeşitli veriler içeren bir veri kaynağıdır.

Seçmeli transfer, transfer edilecek detayların tek tek ya da grup olarak belirtilmesine olanak tanımalıdır. "2222" nolu parselin ya da "95 nolu karayolu" nun belirtilmesi tek tek detay belirtmeye, "Et Balık Kurumuna ait bütün parseller"in, "üç şeritli yollar"ın ya da belli bir coğrafi bölge içerisindeki detayların belirtilmesi de grup halinde detay belirtmeye örnektir. Seçim, örneklerdeki gibi özniteliklere ya da konumsal kapsama göre veya grafik görüntü üzerinden yapılabilir. Dolayısıyla "seçmeli transfer araçları" ile kastedilen, sorgulama dilleri, nesnelerin anılan şekilde belirtmesi için sözdizimler—ve dolayısıyla bunlar için ayrıştırıcılar—, görselleştirme, ve tarama araçlarıdır. Görselleştirme araçları transfer verisinin grafik ekranda gösterilmesi içindir. Tarama araçları ise, belirtilen bir öznitelige göre, nesnelere ait istenen diğer öznitelikleri görmek içindir [40]. Örnek bir uygulama, "Et Balık Kurumu'na ait bütün parseller"in bütün özellikleri ile taranması olabilir.

Tabi seçmeli transfer için, sunucu verisi ya da transfer veri grubunun seçmeli transfer araçlarına uygun şekilde organize edilmiş olması (indekslenmesi vs.) gerekir. Örneğin sorgulama dili olarak SQL in kullanılması, transfer edilecek verinin ilişkisel organizasyonunu gerektirir. Ancak, burada bu konuda daha fazla ayrıntıya girilmeyecektir. Çünkü, seçmeli transfere izin veren bir değişim ortamının nasıl olması gerektiği, aktif bir araştırma konusudur ve bu tezin kapsamını aşmaktadır. Sonuç olarak, DF nin seçmeli transferi olanaklı kılan bir değişim ortamı ile desteklenmesi, standartlaşmasını çok önemli ölçüde teşvik edecektir.

3. BULGULAR

Daha önce de belirtildiği gibi bu tezde amaç, Türkiye için oluşturulması düşünülen Ulusal Konumsal Veri Altyapısı (UKVA) gereksinimlerini karşılayacak bir konumsal veri değişim standardının belirlenmesidir. Tezde benimsenen metodoloji, konumsal veri yönetiminde aşama kaydetmiş ülkeler tarafından geliştirilmiş olan konumsal veri değişim formatlarının belirli ölçütlere göre değerlendirilerek, Türkiye için en uygun birinin seçilmesidir. Bu amaçla belirlenen değerlendirme ölçütleri, bir önceki bölümde tanıtılmıştı. Bu bölümde, değerlendirmeye alınan değişim formatları, değerlendirme ölçütlerine göre incelenmektedir. İrdeleme ve nihai değerlendirme, 4. bölümde ele alınmaktadır.

"Bugün bir kısmı aktif olarak kullanılmakta olan, daha pek çok konumsal veri değişim formatı varken, neden yalnızca bu üç format değerlendirmeye alındı?" diye bir soru akla gelebilir. Gerçekten de bugüne kadar, gerek teknik gerekse politik nedenlerle çok sayıda konumsal veri değişim formatı geliştirilmiştir. Değerlendirmeye alınacak konumsal veri değişim formatlarının belirlenmesi için bu formatlar bir ön elemeye tabi tutulmuştur. Bu formatlardan en çok bilinen bazıları DXF (Drawing Exchange Format) [86], DLG (Digital Line Graph) [87], SIF (Spatial Interchange Format) [88], MACDIF (Map and Chart Data Interchange Format) [80], DX90 [27], NTF (National Transfer Format) [85], TIGER (Topologically Integrated Geographic Encoding and Referencing format) [126] dır. Ayrıca, doğrudan coğrafi veriye yönelik olmamakla birlikte, konumsal veri değişiminde kullanılan ve bazen "CAD standartları" olarak ta anılan pek çok format bulunmaktadır. Bunlara örnek olarak ta, CGM (Computer Graphics Metafile), IGES (Initial Graphics Exchange Specification), TIFF (Tagged Image File Format), GIF, JPEG (Joint Photographers Experts Group), HPGL (Hewlett-Packard Graphics Language) verilebilir.

Dördüncü bölümde söz edildiği gibi, bir konumsal veri değişim formatının geniş kullanım alanı bulması çeşitli sebeplere bağlıdır. Dolayısıyla, yukarıda anılan formatlardan bazıları oldukça yaygın bir kullanım alanı bulmuş, hatta "defacto" standartlar olarak adlandırılmıştır. Bugün bir milyon dolayında kullanıcısı bulunan [127] AutoCAD yazılımının veri değişim formatı olan DXF, defacto standartların en belirgin örneklerden biridir. Diğer bir örnek, A.B.D.'nin başlıca harita verisi sağlayıcı kuruluşu olan Amerikan Jeolojik Ölçmeler (USGS)'in sayısal veri dağıtımında bugüne kadar çok yaygın olarak kullandığı DLG formatıdır. Ancak, ön eleme sonucunda bu alışlagelmiş formatların UKVA'nın

gerektirdiği genellik, esneklik, ve genişletilebilirlikten yoksun olduğu görülmüştür. Çünkü bu formatların çoğu, özellikli amaçlara yönelik olarak geliştirilmiş, oldukça dar kapsamlı formatlardır.

Ön eleme sonucunda, yalnızca üç format değerlendirme ölçütlerine göre değerlendirmeye uygun bulunmuştur. Bunlar Kanada'da geliştirilmiş olan SAIF (Spatial Archive and Interchange Format), A.B.D.'nin SDTS (Spatial Data Transfer Standard), ve NATO'nun DGIWG (Digital Geographic Information Working Group) çalışma grubu tarafından geliştirilmiş olan DIGEST (Digital Geographic Information Exchange Standard) formatlarıdır. Bu formatlar yukarıda anılan alışlagelmiş formatlardan daha sonra geliştirilmiş formatlar olduklarından, yeni bir takım kavramlar içermektedir. Aynı zamanda bunlar, konumsal veri değişiminde gelecekte söz sahibi olacak formatlardır. Bu nedenle, söz konusu üç formatın birbirleriyle uyumluluğunun sağlanması çabaları son birkaç yıldır hız kazanmıştır [23]. Bu bölümde SAIF, SDTS, ve DIGEST değerlendirme ölçütlerine göre incelenmektedir. Bölümde, gerek metin içerisinde ve gerekse şekillerde, değerlendirilen formatlarla özdeşleşmiş terimler (veri tipi isimleri vs.), pratik bir yararı olmayacağı düşüncesiyle Türkçe'ye çevirilmemiş, aynen format dokümantasyonlarındaki gibi İngilizce olarak verilmiştir.

3.1 SAIF (Spatial Archive and Interchange Format)

SAIF (Spatial Archive and Interchange Format), Kanada'nın British Columbia (BC) Eyaleti, "Çevre, Arazi, ve Parklar Bakanlığı"nın, "Ölçme ve Harita Dairesi" (Surveying and Resource Mapping Branch) tarafından geliştirilmiştir. Ocak 1991 tarihli ilk versiyon [40], daha sonra birtakım değişikliklere uğramıştır. Örneğin V2.0 yalnızca tek miras almaya izin verirken, V3.1 ile çoklu miras alma desteği eklenmiştir. Sonuçta Ağustos 1992'de Versiyon 2.0, Ağustos 1994'te Versiyon 3.1 ve onun ardından Versiyon 3.2 çıkarılmıştır. Kanada Genel Standartlar Komitesi SAIF'i 1991'de coğrafi veri değişim ön standardı, daha sonra 1993'te de ulusal standart olarak kabul etmiştir [128]. İzleyen kısımda, SAIF'in bir önceki bölümde tanıtilen her bir değerlendirme ölçütüne göre incelenmesi yapılmaktadır. SAIF'in bir veri değişim ya da arşiv formatı olması yanında bir VTYS nin veri modeli olarak ta kullanılabileceği belirtilmekle birlikte [40], böyle bir gerçekleştirimi henüz yoktur.

SAIF, BC SRMB tarafından 1992'den beri, SRMB verisinin dağıtımında etkin olarak kullanılmaktadır. Yine BC eyaleti Orman Bakanlığı SAIF ARC/INFO ve Intergraph formatları için çeviriciler geliştirmektedir. SAIF ayrıca Kanada'nın Saskatchewan eyaleti, Ölçme ve Harita Dairesi tarafından ve Californiya Üniversitesi'nin "Sequoia 2000" projesinde kullanılmaktadır [35].

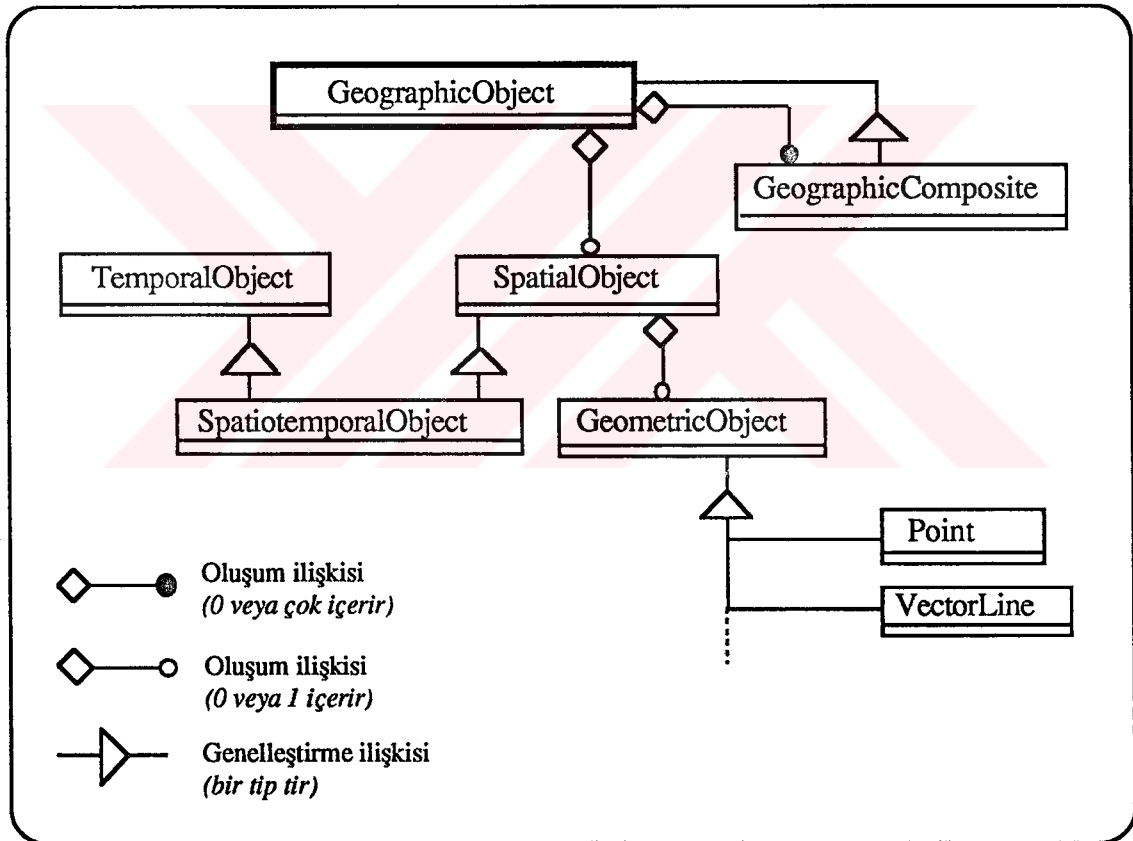
3.1.1 Veri Modeli

SAIF, Nesneye Dayalı (ND) bir veri modeline dayanmaktadır. SAIF veri modeli oldukça genel ve kapsamlıdır. Drenaj ağlarından, kömür yataklarına çok farklı varlıklar tanımlanabilir. Bununla birlikte SAIF, coğrafi detaylar için yalnızca bir "alt yapı" sunmakta, özellikli detay tanımlama ise bu alt yapı üzerine kurulmaktadır. Diğer bir ifadeyle, detay tanımlama SAIF kapsamı dışında tutulmuş, yani kullanıcılara (gerçekleştirimcilere) bırakılmıştır. İkinci bölümde de değinildiği üzere, farklı kullanıcılar aynı detayı farklı şekillerde tanımlayacakları için, bu tanımlamaların standartlaştırılması çok zor olacaktır. Bu açıdan SAIF tasarım felsefesi "farklı detay tanımlamaları arasında geçiş sağlanabildiği sürece bırakalım herkes detayları kendi istediği, kendine en uygun şekilde tanımlasın" demektedir. Bu ilke, SAIF'i DIGEST ve SDTS'ten ayıran en belirgin özelliklerden biridir. Formatın özellikli transferler için uyarlanabilmesi, yani esneklik açısından da SAIF kullanıcılara çeşitli seçenekler sunmaktadır. Örneğin, alışlagelmiş CBS'ler arasında veri transferi söz konusu olduğunda, SAIF'in özellikli alt-grupları (profiller) kullanılabilir ve özellikli transferlerde genel bir model kullanılmasının yolaçacağı verimsizlik önlenir. SAIF aynı zamanda, çok yüksek oranda bir genişletilebilirlik sunmakta ve gerektiğinde SAIF standart şeması genişletilebilmektedir. Diğer bir anlatımla, miras devralma yoluyla SAIF şemasında yer alan sınıfların özellikli alt-sınıfları tanımlanabilmektedir [35].

3.1.1.1 Veri Modeli Tipi

SAIF, ND veri modeline dayanmakta ve çoklu miras almayı desteklemektedir. Veri soyutlama mekanizmaları genelleştirme/özelleştirme, birleştirme, ve ilişkilendirme desteklenmektedir [35]. SAIF Standart Şeması 300'ün üstünde veri tipi (sınıf) tanımını içermektedir. Veri tipi tanımlamaları yalnızca veri yapısını belirtmekte, işlemleri (rutinler) kapsamamaktadır. SAIF Standart Şemasının en önemli sınıfı "coğrafi nesne" (GeographicObject) (Şekil 29) dir. Yeryüzü üzerindeki akarsu, karayolu, parsel gibi varlıkların her biri birer coğrafi nesne olarak görülür. Bununla birlikte, bir "coğrafi nesne" nin bir akarsu ya da bir karayolu mu olduğu SAIF veri modeli ya da şemasında belirtilmez. Farklı SAIF kullanıcıları özellikli varlık tanımlamalarını coğrafi nesne'nin altında "alt sınıf" tanımlama yoluyla yaparlar [35]. Dolayısıyla coğrafi nesne, özellikli varlık tanımlama için bir kalıp oluşturmaktadır. Bu yolla, "gerçek içerik" SAIF kapsamı dışında tutulmuş, onun yerine gerçek içeriği tanımlamaya uygun bir iskelet yapı tanımlanmıştır. Bu, farklı kullanıcılar açısından bakıldığında oldukça uygun olmaktadır. Çünkü farklı kullanıcılar gereksinimleri gereği varlıkları farklı şekillerde tanımlamak durumundadırlar. Bu belki de, SAIF'i alışlagelmiş DF'lerden ayıran en önemli özelliktir.

Bir Coğrafi nesne, konumsal ve konumsal olmayan öznitelikleri ile tanımlıdır. Konumsal öznitelikler için "SpatialObject" sınıfı tanımlanmıştır. Buna göre bir coğrafi nesnenin konumu ve geometrisi bir SpatialObject nesnesi ile tanımlanır. Bir SpatialObject 0 veya 1 "GeometricObject" nesnesini içerebilir (Şekil 29). Nokta, çizgi, poligon gibi temel geometrik elemanlar GeometricObject sınıfı altında tanımlanmıştır. Buna göre bir nokta, bir tip Geometrik nesnedir (daha özel bir tip). Konumsal olmayan öznitelikler ise, coğrafi nesnenin alt-sınıfı olarak tanımlanacak varlık sınıfları içerisinde, istenen şekilde tanımlanabilir. SAIF'te Coğrafi nesnelerin birbirleriyle olan ilişkilerinin gösterimi için, "Relationship" sınıfı bulunmaktadır. Coğrafi nesneler arasındaki konumsal ve zamansal ilişkiler Relationship sınıfı kapsamındadır. Böylece, konumsal ve zamansal ilişkiler, alışlagelmiş veri modellerinde olduğundan daha yüksek bir düzeyde (coğrafi nesne düzeyinde) gösterim edilmektedir.



Şekil 29. SAIF'te GeographicObject kavramı.

3.1.1.2 Raster Veri

SAIF'in raster veri desteği oldukça güçlüdür. Raster veri için, "Raster" sınıfı bulunmaktadır. Raster veri yapılarının konumsal ve konumsal-zamansal olarak tanımlanabilmesi için, Raster sınıfı "CelularStructure" sınıfı ile ilişkilendirilmiştir. Bu yolla,

yükseklik ve zaman boyutlarının da belirtilmesi olanaklı olmaktadır. Farklı raster veri formatları için, Raster sınıfı altında "ImageRaster", "BlockRaster" gibi çeşitli sınıflar tanımlanmıştır. Bu sınıflar yardımıyla raster veri, band sequential, band interleaved, run length encoded (RLE), ve dörtlü-ağaç gibi çok yaygın kullanılan formatlarda kodlanabilir.

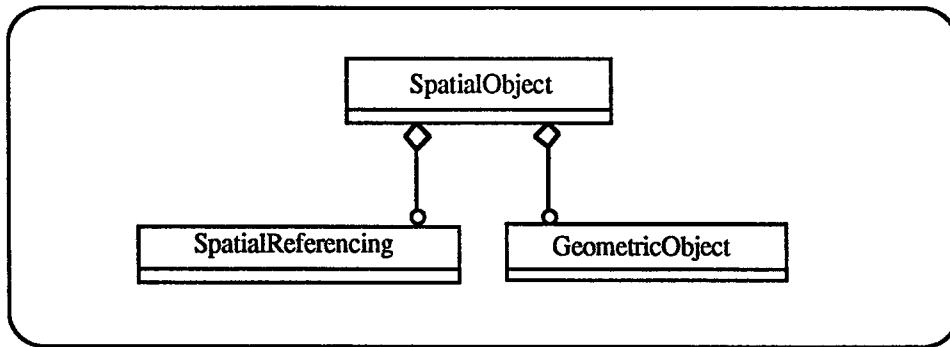
3.1.1.3 Kartoğrafik Gösterim

SAIF'te metin ve işaretler için "TextOrSymbolObject" sınıfı bulunmaktadır. Akarsu isimleri gibi, eğri çizgiyi izleyen metin gösterilebilir. Diğer bir anlatımla, bir metni oluşturan her bir karakter tek tek, genişliği, yüksekliği vs. ile belirtilebilir. SAIF bunun için, CGM (Computer Graphics Metafile) modelini kullanmaktadır. İşaretler için TextOrSymbolObject sınıfı altında ayrıca "Symbol" sınıfı tanımlanmıştır. Böylece bir işaretin konumu, ait olduğu işaret kitaplığı, ve bu kitaplıktaki kimliği belirtilebilir.

SAIF standart şeması, UTM, Mercator, Lambert Konformal, Polyconic ve daha bir dizi, yaygın kullanım bulmuş projeksiyon tanımını içermektedir. Bunların yeterli olmaması durumunda kullanıcı kendi projeksiyon sistemini tanımlayabilir.

3.1.1.4 Koordinatlar

SAIF, koordinatların iki (x, y), üç (x, y, z), ve dört boyutlu (x, y, z, t) olarak kodlanmasına izin verir. Ayrıca bu dört boyutun değişik kombinasyonları, örneğin yalnızca x , t kullanılabilir [35]. Ayrıca kullanıcılar, gerektiğinde amaçlarına özel alt-sınıflar tanımlayabilirler. Koordinat sistemleri ve referans sistemlerinin belirtilebilmesi için "SpatialReferencing" sınıfı bulunmaktadır. Böylece, yatay ve düşey referans sistemleri yanında, transformasyon parametreleri de belirtilebilir. SpatialReferencing sınıfı, SpatialObject sınıfı ile ilişkilendirilmiştir (Şekil 30). SAIF'te yatay dengeleme sistemleri,



Şekil 30. SAIF'te "SpatialObject" sınıfı.

yalnızca Kanada ile sınırlı değildir. Benzer şekilde, SAIF standart şemasında düzey dengeleme sistemi yalnızca Kuzey Amerika için belirtilmişse de, gerektiğinde yeni sistemler için tanımlamalar eklenebilir [35]. SAIF standart şemasında 19 projeksiyon tipi tanımlıdır. Bunların yeterli olmaması durumunda kullanıcı kendi projeksiyon tipini tanımlayabilir.

Koordinatlar için tamsayı ("Integer") kullanılması durumunda 64 bit, kesirli sayı ("Real") kullanılması durumunda da 80 bit lik bir çözünürlükle kodlama yapılabilir [129].

3.1.1.5 Topoloji-Geometri Ayırımı

SAIF, topoloji-geometri ayırımı kavramını desteklemektedir. SAIF'te topoloji, "Relationship" sınıfı altında tanımlanmıştır. Coğrafi nesnelerin birbirleriyle olan konumsal ve zamansal ilişkileri Relationship sınıfının alt sınıfları yardımıyla gösterim edilir. Diğer bir anlatımla herhangi iki coğrafi nesne arasında bir ilişki, bir Relationship nesnesi ile gösterim edilir. Eğer Relationship nesnesi yoksa, sözkonusu ilişki ilgili veri grubunda gizli kalacak, yani gösterim edilmeyecektir. Dolayısıyla, istendiğinde topoloji kapalı tutulabilir ve yalnızca metrik veri transfer edilebilir.

3.1.1.6 Karmaşık Nesneler

SAIF'in karmaşık nesne desteği vardır. Karmaşık nesneler için "GeographicComposite" sınıfı bulunmaktadır. Bir GeographicComposite nesnesi diğer GeographicObject nesneleri içerir (Şekil 29). Buna göre örneğin, tek tek "karayolu" nesnelerinden bir "karayolu_ağı" oluşturulabilir. Şekil 29'dan anlaşılabacağı gibi, GeographicComposite tipi nesneler aynı zamanda, GeographicObject tipinde nesnelerdir. Böylece özyineleme olanaklı olmakta, bir GeographicComposite nesnesi başka GeographicComposite nesneler içerebilmektedir. Buna göre bir karayolu_ağı tek tek karayollarından oluşabileceği gibi, karmaşık nesneler olan karayolu ağlarını da yapısında bulundurabilir. Dolayısıyla, SAIF karmaşık nesnelerin modellenmesinde oldukça yüksek bir esneklik sunmaktadır.

3.1.1.7 Nesne Paylaşımı

SAIF nesne paylaşımını desteklemektedir [35]. Nesne kimliği vasıtasıyla, farklı nesneler aynı nesneye gönderimde bulunabilir. Bu yolla örneğin, iki poligon arasında sınır oluşturan bir çizgi parçası, yalnızca bir kez kodlanır ve her iki poligon da bu çizgi parçasının nesne kimliğine işaret edebilir. Nesne paylaşımı yalnızca geometrik nesne bazında değil aynı zamanda coğrafi nesne bazında da olabilir. Buna göre, iki farklı karmaşık nesne aynı coğrafi

nesneye işaret edebilir (iki farklı karayolu ağının aynı karayolu nesnesini kullanması gibi). Nesne paylaşımı "2" nolu SAIF profili tarafından desteklenmektedir [84].

3.1.1.8 Meta Veri

Meta veri için, SAIF'te "MetaData" sınıfı bulunmaktadır [35]. SAIF standart şemasında, MetaData sınıfının altı alt sınıfı bulunmaktadır. Bunlardan bazıları "SpatialReferencing", "Source", "Quality" dir. Bu sınıflar yardımıyla, verinin yatay ve düşey referans sistemleri, kaynağı, doğruluğu vs. belirtilebilir. SAIF standart şemasındaki meta veri tanımlamalarının yeterli olmaması durumunda kullanıcı, gerek MetaData sınıfının ve gerekse onun alt sınıflarının daha özellikli alt sınıflarını tanımlayabilir. Böylece meta veri belirtmede çok yüksek bir esneklik sunulmaktadır. SAIF'te meta transfer veri grubunun tamamı için genel olarak ya da tek tek nesneler düzeyinde belirtilebilir.

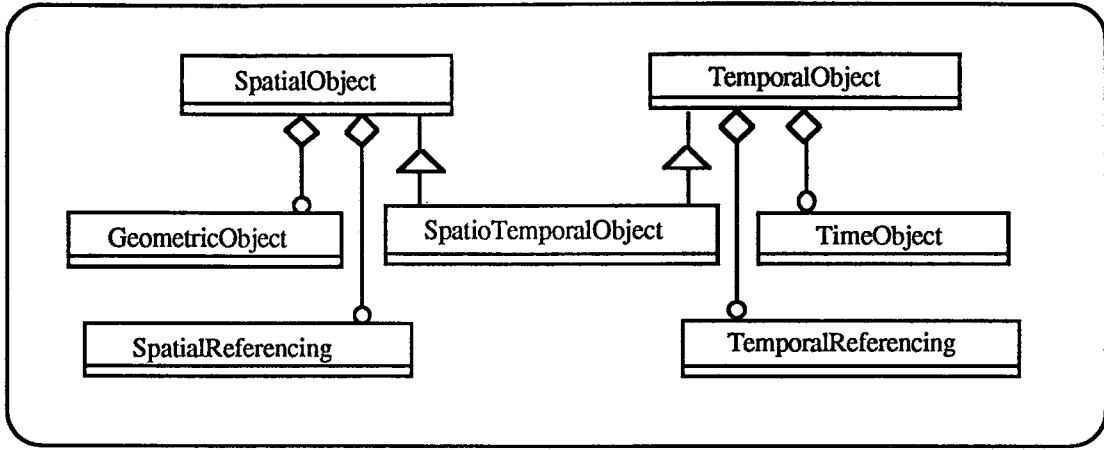
3.1.1.9 Nesne Bazında Veri Değişimi

SAIF nesne bazında veri değişimini desteklemektedir. Bu zaten SAIF'in başlıca tasarım hedeflerinden biri olmuştur [35]. SAIF ile veri transferinde transfer veri grubu, bütün bir dosya ya da tek bir nesneye ait veriyi içeren bir veri dizisi olabilir.

3.1.1.10 Zamansal Veri

Yukarıda da belirtildiği gibi SAIF, zamanın da bir koordinat olarak belirtilmesine olanak tanımaktadır. Bunun yanında, SAIF'te zamansal veri için "TimeObject", "TemporalObject", "SpatiotemporalObject", ve "TemporalReferencing" sınıfları ile, bunların alt sınıfları bulunmaktadır. TimeObject sınıfı yardımıyla saat, tarih, zaman aralığı, ve süre belirtilebilmektedir. TemporalReferencing ise, zamansal veri için referans sitemini (örn. UTC (Universal Coordinated Time), yerel vs.) belirtmek içindir. SpatiotemporalObject sınıfı, SpatialObject ve TemporalObject sınıflarının alt sınıfı durumundadır (Şekil 31).

SAIF'te nesnelerin birbirleriyle olan zamansal ilişkilerinin belirtilebilmesi için "TemporalRelationship" sınıfı ile onun alt sınıfları (örn. "TemporalTopology", "TemporalNeighbourhood") tanımlanmıştır. Standart şemada yer alan bu alt sınıflar yanında, kullanıcılar amaçlarına uygun olarak TemporalRelationship'in özellikli alt sınıflarını da tanımlayabilirler. Sonuç olarak, SAIF'in zamansal veri desteği oldukça güçlüdür.



Şekil 31. "SpatiotemporalObject" sınıfı ve diğer ilgili sınıflar.

3.1.1.1 Multimedya

SAIF, multimedya desteği sunmamaktadır.

3.1.1.12 Genişletilebilirlik

SAIF, Nesneye-Dayalı bir veri modeline sahip olması nedeniyle yüksek derecede bir genişletilebilirlik sunmaktadır. SAIF standart şeması 300 den fazla sınıf tanımını içermekte ve gerektiğinde bu sınıflardan yeni/özellikli sınıflar tanımlama için gerekli alt yapıyı sağlamaktadır. Diğer bir anlatımla SAIF, eldeki bir uygulama ya da veri transferi için, kullanıcılara kendi özellikli sınıflarını tanımlamaları için genelleştirme/özelleştirme, birleştirme, ilişkilendirme gibi mekanizmaları ve ilgili kuralları sunmaktadır. Böylece farklı kullanıcılar amaçlarına uygun özellikleri taşıyacak şekilde coğrafi detayları "GeographicObject" sınıfının alt sınıfı olarak tanımlayabilirler. Benzer şekilde karmaşık coğrafi detay tanımlamaları için "GeographicComposite" sınıfından çok önemli ölçüde yararlanabilirler. Ya da örneğin, SAIF standart şemasında bulunmayan bir projeksiyon sistemi için yeni bir sınıf tanımlayabilir.

Farklı kullanıcılar amaçlarına uygun olarak aynı coğrafi detayları farklı şekillerde tanımlayabilirler. Farklılık konumsal ve konumsal olmayan öznitelikler bakımından olabilir. Örneğin, bir "parsel" tanımlamasında parsel sınırları çizgi parçalarından oluşmakta iken, diğerinde noktalardan oluşabilir. Ya da bir "karayolu" detayının "yol kaplaması" bir kullanıcı için çok önemli, diğer bir kullanıcı için ise tamamen anlamsız olabilir. Farklı kullanıcıların aynı coğrafi nesneleri farklı şekillerde tanımlamaları sonucunda, farklı şemalar ortaya çıkacaktır. SAIF bu bakımdan, yüksek derecede bir esneklik ve genişletilebilirlik sunmaktadır. Çok farklı şemaların tutarsızlık yaratmadan birarada bulunabilmesi,

interoperabilite açısından son derece önemlidir. Bu bakımdan SAIF, "ortak" (common/canonical) bir veri modeli sunmaktadır [130].

3.1.2 Gerçekleştirim

SAIF, ND gerçekleştirimi desteklemektedir. İkinci bölümde belirtildiği üzere, ND veri modelinde veri yapısı yanında çeviri, kodlama, kod çözümleme, ve hatta iletişim rutinleri nesneler içerisinde enkapsüle edilebilir. Ancak SAIF Standart Şeması, söz konusu rutinleri içermemekte bunların eklenmesini gerçekleştirimciye bırakmaktadır [35]. SAIF, genel bir format olduğu için gerçekleştirimde profil kullanımını öngörmektedir. Detay kodlama açısından SAIF'in kendine ait bir detay kodlama katalogu yoktur. Çünkü detay tanımlama SAIF kapsamı dışında tutulmuştur. Bununla birlikte, SAIF detay kodlama standartlarıyla (örn. FACC) birlikte kullanılabilir [79]. Fiziksel kodlamada ise SAIF/ZIP kullanılmaktadır. SAIF/ZIP 3.1.2.3'te özetle tanıtılmaktadır.

3.1.2.1 Profiller

SAIF'in oldukça esnek ve genişletilebilir bir veri modeli sunması bir avantaj olması kadar bir dezavantaj olarak ta görülebilir. Çünkü SAIF kullanılarak aynı veri çok çeşitli şekillerde tanımlanabilir. Öte yandan, SAIF veri modeli alışlagelmiş CBS'lerinkinden daha kapsamlıdır [130]. O nedenle belirli bir veri transferi için SAIF'in olduğu gibi kullanılması yerine, özellikli bir alt grubunun kullanılması işlemsel verimliliği önemli ölçüde artıracaktır.

SAIF'in özellikli bir alt grubu, bir "SAIF profili" olarak anılır. Profiller belirli kural ve sınırlamalara göre oluşturulur [35]. Örneğin yalnızca vektör veriye yönelik bir profil tanımlanabilir. Bu profil poligonları ve yalnızca belirli çizgi ve nokta tiplerini tanıyabilir, koordinatlarda zaman boyutunu belirtmeye izin vermeyebilir. Hatta verimi artırmak için transfer veri grubu içerisinde yer alan nesnelerin sırası da bir profil özelliği olarak belirtilebilir.

SAIF profillerinin tanımlanması ve yaşatılmasının belirli bir otorite tarafından yapılması gerekir. Bu otorite şimdilik, SAIF'i geliştiren SRMB çalışma gurubudur. Bu grup bu güne kadar dört ayrı SAIF profili öngörmüştür. Bunlardan ilki tamamlanmış, diğer üçü geliştirilme aşamasındadır [84]. Tamamlanmış ve kullanımda olan SAIF profili, coğrafi veriyi çoğu alışlagelmiş CBS'de tutulduğu haliyle gösterim etmek üzere tanımlanmıştır. Diğer bir anlatımla, bu profil alışlagelmiş CBS lerin büyük bir çoğunluğu arasında veri değişimine yöneliktir. Bu profil, nesne paylaşımı ve nesneler arası konumsal ve zamansal ilişkilerin belirtilmesine izin vermemektedir. Bu tür ilişkiler içeren verinin gösterimi, 2 ve 3 nolu SAIF profilleri kapsamındadır. 4 nolu SAIF profili ise, raster veriye yöneliktir [84].

Ayrıca, SAIF ve DIGEST (Digital Geographic Exchange Standart) arasında geçişi sağlamak üzere bir DIGEST profili geliştirilmektedir [23]

SAIF formatında veri belirli bir "kullanıcı profili"ne uygun olarak gelir. Bir kullanıcı profili, belirli bir SAIF profili üzerine o kullanıcı grubuna ait özellikli veriyi tanımlamak üzere oluşturulur. Kullanıcı profillerinin belli bir otorite tarafından tanımlanması gerekmez. Her kullanıcı grubu kendi verisi için bir profil tanımlayabilir. Farklı coğrafi veri kullanıcıları, veriyi farklı şekillerde gördüklerinden, çok çeşitli kullanıcı profilleri ortaya çıkacaktır. Bu nedenle bir SAIF çeviricisinin bu kullanıcı profillerini tanıyabilmesi için, transfer veri grubunu tanımlayan SAIF profilleri ve kullanıcı profilleri ile bunların versiyon numaraları, ve tanımlayıcı otorite transfer veri grubunun bir parçası olarak belirtilir.

3.1.2.2 Detay Kodlama

Yukarıda belirtildiği gibi, coğrafi detay tanımlama SAIF kapsamı dışında tutulmuştur. SAIF yalnızca farklı kullanıcıların kendi amaçlarına uygun olarak coğrafi detayları tanımlayabilmeleri için gerekli temel yapıyı sunmaktadır. Özellikle coğrafi detay tanımlamalarının bu temel yapı üzerine inşa edilmesi, tamamen kullanıcılara bırakılmıştır. Bunun nedeni çok geniş bir kullanıcı yelpazesine sahip olan coğrafi verinin, standart bir katalog halinde sınıflandırılmasının ve bu katalogun yaşatılmasının oldukça güç olacağı ve sonuçta oluşacak katalogun çok kapsamlı olabileceğidir [79]. Bu nedenle SAIF, herhangi bir standart detay/öznitelik kodlama katalogu (FACC gibi) ile birlikte düşünülmemiştir. Bununla birlikte, istendiğinde böyle bir katalog SAIF ile birlikte kullanılabilir [79]. Böylelikle kullanıcılara çok yüksek bir esneklik sunulmaktadır. Ancak sözkonusu esneklik, doğal olarak bazı tutarsızlıkları beraberinde getirecektir. Çünkü örneğin, bir imar planı ile bir harita mühendisinin parsel tanımı birbirinden farklı olacaktır. Farklı tanımlamaların bir arada bulunabilmesi sorunu, farklı tanımlamaların bilinmesi ve aradaki dönüşümün bir şekilde sağlanmasıyla çözülebilir. Bu amaçla bir SAIF veri setinde tanımlamayı içeren şema adı belirtilir [35].

SAIF, aslında tasarım felsefesi bakımından, alışlageldiği şekliyle detay kodlamaya terstir. Çünkü, SAIF ND veri modeline dayanmakta ve ND gerçekleştirimi desteklemektedir. Gerçi ND modelde de veri, detay kodlamada olduğu gibi öznitelikler itibarıyla sınıflandırılır ancak, ND modellemede sınıflandırma öznitelik değerlerinden ziyade doğrudan öznitelikleri dikkate alır. Alışlagelmiş detay kodlamada, farklı öznitelik değerlerinin herbiri ve bu değerlerin her bir farklı kombinasyonu için ayrı detay kodlarının oluşturulması gerekir (bkz. 1.2). Gerçekleştirmede, bu kodların ilgili yazılıma—veri değişiminde çevirici—tanıtılması gerekir. ND yöntemde ise sözkonusu kodların oluşturulması ve belki daha önemlisi zamanla

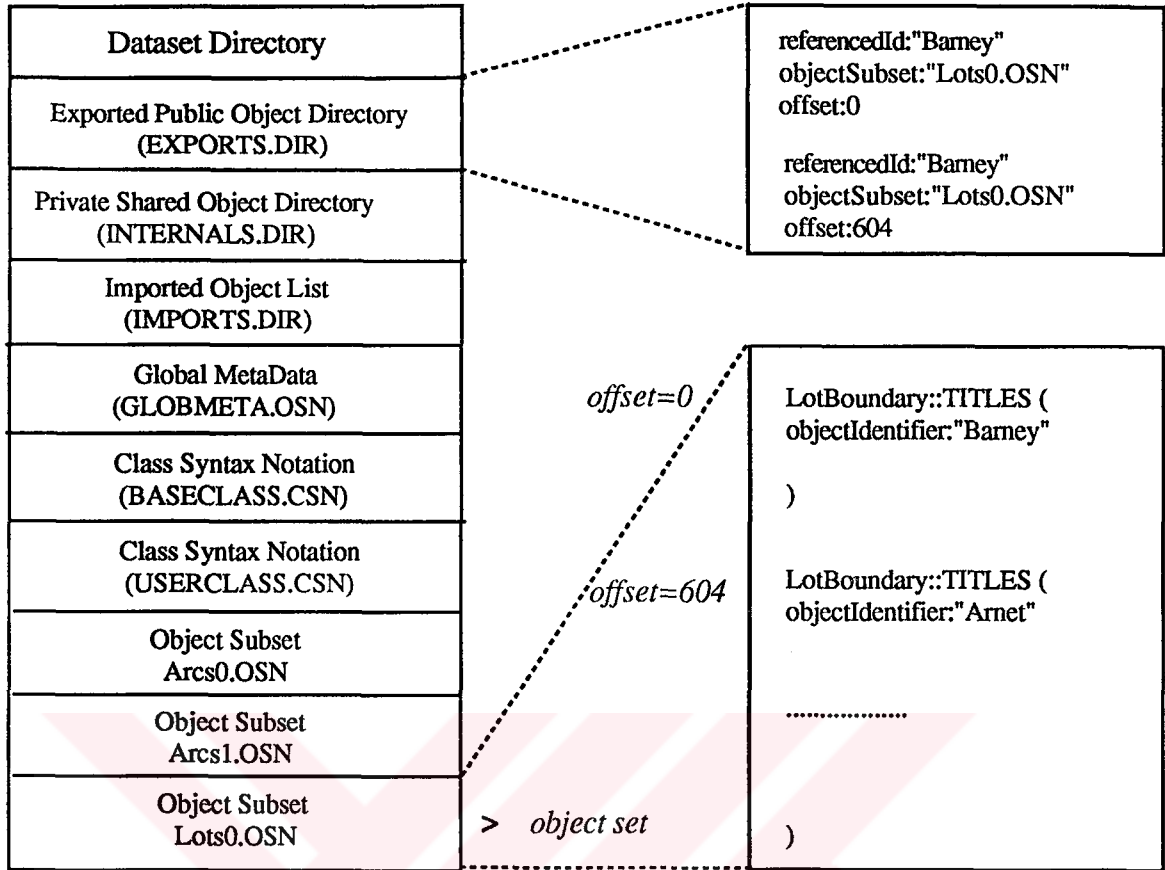
yenilenmesi yükü çok önemli ölçüde kullanıcıların sırtından alınmıştır (bkz. 2.2.2). Gerçi gerçekleştirim açısından öznitelik değerlerinin kodlanması gerekir ancak bu, sınıflandırma hiyerarşisi içerisinde her sınıfta bağımsız olarak yapılan, detay kodlamaki gibi sınıflandırma hiyerarşisine bağlı olmayan çok daha basit bir kodlamadır. Buna göre ND bir gerçekleştirimde, detay kodlamaya dayalı bir gerçekleştirimdekine eşdeğer kodlar, istendiğinde otomatik olarak türetilir. Ayrıca, ND yöntemde kodların sınıflandırma hiyerarşisinden bağımsız olarak sınıf bazında verilmesi, kodlamanın genişletilebilirliği açısından da son derece pratiktir; Zamanla gündeme gelebilecek ya da başlangıçta düşünülemeyen öznitelik değerleri için basitçe yeni bir kod eklenmesi yeterli olabilecekken, detay kodlamada yeni öznitelik değerinin mevcut kodlamaya etkisi çok daha fazla olabilir. O nedenle, ND yöntemde sınıflandırmanın ayrıntı düzeyinin belirlenmesi, detay kodlamadaki kadar kritik değildir.

3.1.2.3 Kodlama Yöntemi

SAIF 2.0, fiziksel kodlama için Sun Microsystems tarafından geliştirilen XDR (eXternal Data Representation) kodlama kurallarını kullanmakta idi. SAIF 3.1'de ise, SAIF/ZIP kodlama yöntemi benimsenmiştir. SAIF/ZIP'e göre SAIF verisi, sıkıştırılmış ASCII OSN (Object Syntax Notation) formatında gelir [131]. SAIF/ZIP formatı, başka formatlardaki verinin SAIF verisi ile birlikte transferine ya da arşivlenmesine izin vermektedir.

"SAIF veri grubu" ile, ".ZIP" uzantısı taşıyan dosya anlaşılmalıdır. Bir "nesne grubu", sıralı ASCII OSN dosyalarından oluşur ve bir ya da daha çok nesneyi içerir. Bir SAIF veri grubu içinde, birden çok nesne grubu bulunabilir (Şekil 32). Bir nesne grubu, oldukça geniş olabilir ve nesne alt gruplarından oluşur. Bir "nesne alt grubu", SAIF veri seti içerisinde adreslenebilir bir birimdir. Nesne alt grupları ".OSN" uzantısını taşır. Örneğin bir SAIF veri grubu, YOLLAR isimli bir nesne seti içerebilir; Bu nesne grubu da YOLLAR0.OSN, YOLLAR1.OSN gibi alt gruplardan oluşabilir. Şekil 32, SAIF veri grubunun yapısını göstermektedir.

Her bir SAIF veri grubu, bir ya da daha çok CSN (Class Syntax Notation) dosyası içerir. Bunlar, nesne grubunda yer alan nesnelerin sınıflarını tanımlar. Bir SAIF veri grubu, gerçek veri yanında verinin tanımını da içerir. SAIF veri grubunun, SAIF standart şemasında yer alan sınıf tanımları BASECLAS.CSN dosyasında bulunur. ".OSN" uzantılı dosyalar ise, transfer edilen nesnelerin OSN tanımlarını içerir. GLOBMETA.OSN dosyası, SAIF veri grubunun tümü için geçerli olan meta veriyi içerir. EXPORTS.DIR dosyasında, SAIF veri grubu ile ihraç edilen nesnelerin bir listesi yer alır. IMPORTS.DIR'de ise ithal nesne listesi



Şekil 32. SAIF veri grubunun yapısı.

bulunur. Her bir SAIF veri grubunda, en az bir EXPORTS.DIR bulunmalıdır. IMPORTS.DIR için böyle bir zorunluluk yoktur. INTERNALS.DIR dosyası veri grubunun paylaşımına konu olan nesnelerinin bir listesini kapsar. Nesne paylaşımı söz konusu değilse, bu dosyaya gerek yoktur.

SAIF/XDR'da etiketli ve ikili bir kodlama uygulanmakta iken, SAIF/ZIP'te etiketsiz, ASCII kodlama uygulanmaktadır. Transfer veri grubu SAIF/ZIP'te, SAIF/XDR'a göre daha az kayıt alanı gerektirir. Bunun nedenlerinden biri, SAIF/XDR'da etiketlerin çok yoğun olarak kullanılması, diğeri ise SAIF/ZIP'te sıkıştırma uygulanmasıdır. Sıkıştırma çok yaygın olarak kullanılan bir yöntemle yapılmaktadır. SAIF/ZIP, gerek çevrim içi ve gerekse çevrim dışı veri transferi için uygundur [131].

3.1.3 Değişim Ortamı

"SAIF Toolkit" [132], Kasım 1994'de kullanıma açık—ftp edilebilir—olarak piyasaya sürülmüş ve halen gerekli olabilecek iyileştirmeler için, çeşitli kullanıcı kesimlerinin inceleme ve yorumları beklenmektedir [128]. SAIF Toolkit, SAIF çeviricilerinin geliştirilmesini

önemli ölçüde kolaylaştıracak olan, bir "C" rutin kitaplığı ve çeşitli ayrıştırıcılar ile SAIF şemalarının incelenmesi ve test edilmesi için gerekli araçları içermektedir. SAIF Toolkit, "C++" dilinde yazılmış olmakla birlikte, çevirici geliştiriciler isterlerse "C" dilini kullanabilirler. SAIF Toolkit, Unix ve Microsoft Windows ortamlarında kullanılabilir. Ayrıca, SAIF Toolkit'ini tamamlayıcı nitelikte bir "SAIFTools" destekleyici yazılım paketinin 1996 başlarında tamamlanması planlanmıştır [128].

3.1.3.1 Dokümantasyon

Özellikle SAIF 3.2 dokümantasyonu oldukça iyidir. Kavramlar iyi açıklanmış, veri modeli ve gerçekleştirim arasındaki ayırım özellikle vurgulanmıştır. Kullanılan terminoloji standart terminolojidir. Aslında SAIF 2.0 dokümantasyonunda bir takım eksiklik ve yetersizlikler vardı. Bunlar SAIF 3.2'de önemli ölçüde giderilmiştir. Örneğin sınıflar (nesneler) arası ilişkileri açıklayıcı şekiller konmuştur. Bunun için kullanılan SAIF/OMT (Object Modelling Tool) grafik sözdizimi, gerek SAIF standart şemasını oluşturan ve gerekse yeni eklenen sınıflar arasındaki ilişkilerin grafik gösterimi ve böylece şemaların daha iyi anlaşılması açısından önemli bir yardımcı araçtır. SAIF 3.2 versiyonu kullanıma açıktır. Yani internet aracılığı ile ücretsiz olarak elde edilebilir.

3.1.3.2 Servisler

SAIF veri modelinde servisler bulunmamaktadır. Diğer bir anlatımla SAIF tasarımı veri yapısı üzerinde yoğunlaşmış, veriye uygulanabilecek işlemler SAIF kapsamı dışında tutulmuştur. Bunun gerekçeleri olarak işlemlerin veri değişiminde gerekli olmadığı, işlemlerin tanımlanmasının karmaşık bir iş olacağı, ve istendiğinde gerekli işlemlerin rahatlıkla eklenebileceği gösterilmiştir. Ayrıca işlemlerin veri modeline eklenmesiyle veri transferi sırasında ilgili işlemlerin de transfer edilmesinin gerekeceği ve bunun arzu edilen bir durum olmadığı belirtilmiştir [35]. Bu gerekçelerden son üçü geçerli olmakla birlikte, ilkinin katılmak güçtür. Çünkü yazara göre ve [81] tarafından da vurgulandığı üzere, çevirici rutinlerin bir veri değişim formatı kapsamında düşünülmesi çevirici geliştirmeyi çok önemli ölçüde kolaylaştıracaktır. ND veri modeli bu açıdan idealdir (bkz. 1.3). Çünkü, nesne tanımlamaları (sınıflar) içerisinde, gerek veri yapısı ve gerekse işlemler (çeviri rutinleri) enkapsüle edilebilir. Böylelikle aynı çeviri rutinleri, farklı veri değişimleri için tekrar tekrar kullanılabilir ve çeviriciler çok daha kısa zamanda geliştirilebilir. Çevirici rutinlerin ilgili nesnelere bağlanması, interoperabilite açısından da oldukça geçerli bir yaklaşımdır. Çünkü "federe CBS ler" çevirilerin "anında" yerine getirilmesini gerektirir. Bu da, nesnelere ilgili çeviri rutinlerinin uygulanması ile gerçekleştirilebilir.

SAIF veri modeli çeviri rutinlerini içermemekle beraber, istendiğinde gerek çeviri rutinleri ve gerekse diğer işlemler—SAIF'in genel amaçlı ya da ortak bir veri modeli olarak kullanılması durumunda—eklenebilir [35]. Çeviri rutinlerinin yalnızca bir kaç veri modeline bağlı olmayıp, mümkün olduğunca çok sayıda gönderici ve alıcı modelleri arasındaki çevirileri sağlayacak şekilde belirtilmesi ile bir "süper model" elde edilebilir [81]. Ancak bütün olası gönderici ve alıcı veri modelleri için yeterli olacak çeviri rutinlerinin belirlenmesi ve bunların organizasyonu oldukça karmaşık bir iş olacaktır. O nedenle başlangıçta, yalnızca çok yaygın olarak kullanılan veri modelleri dikkate alınabilir ve gereksinim duyulduğunda yeni rutinler eklenebilir. Bu bakımdan süper modelin aynı zamanda genişletilebilir olması gerekir.

SAIF çeviricilerinin geliştirilmesini kolaylaştırmak amacıyla geliştirilmiş olan SAIF Toolkit, SAIF formatında veri okunabilmesi ve yazdırılabilmesi için gerekli rutinleri içermektedir. Ancak bu rutinler ND veri modelinin ya da yukarıda söz edilen süper modelin öngördüğü şekilde, nesneler içerisinde enkapsüle edilmiş değildir.

3.1.3.3 Değişim Ortamı Araçları

Derleyici ve yorumlayıcı araçları : SAIF, derleyici/yorumlayıcı teknolojisi ile SAIF çeviricilerinin geliştirilebilmesi için gerekli desteği sunmaktadır. SAIFToolkit, çeşitli ayrıştırıcılar içermektedir. Bunlardan örneğin CSN ayrıştırıcısı aracılığı ile CSN'deki sınıf tanımlamalarından, örneğin C++ programlama dilindeki, sınıf kodu türetilir. Böylece kullanıcıların özellikli sınıf tanımlamaları için yapması gereken, yalnızca bu tanımlamalarını CSN'i kullanarak belirtmektir. Bu, kullanıcılar açısından çok önemli bir kolaylıktır. SAIFToolkit'te yer alan diğer bir ayrıştırıcı da, CSV (Comma Separated Value) ayrıştırıcısıdır. CSV formatı, veri elemanı değerlerinin virgüllerle birbirinden ayrıldığı, "Spreadsheet" türü yazılımların çıktı formatıdır. Bir SAIF çeviricisinin geliştirilmesinde alıcı ve göndericinin kullandıkları Detay Kodları (DK) arasındaki ilişkiler CSV formatında bir dosya ile belirtilir. CSV ayrıştırıcısı bu dosyadan bir "Detey Kodu Korelasyon Tablosu (DKKT)" oluşturur ve bu DKKT çevirci tarafından kullanılır. Bu araçlar TRIM/MOEP formatındaki SRMB/BC verisinin, SAIF/XDR formatına dönüştürülmesi projesinde [133], çeviricinin geliştirilmesinde kullanılmıştır. Bunun yanında, SAIF çeviricilerinin bir yorumlayıcı kullanarak geliştirilmesine yönelik çalışmalar devam etmektedir [134].

Veri Tanımlama Dili (VTD) : SAIF VTD olarak "SAIFtalk" bulunmaktadır. SAIFtalk'ın iki bileşeni vardır. Bunlar "CSN" (Class Syntax Notation) ve "OSN" (Object Syntax Notation) dir. Dolayısıyla, "SAIFtalk = CSN + OSN" şeklinde düşünülebilir. CSN, SAIF standart şemasında yer alan sınıfların özlü bir biçimde tanımlanması için kullanılır.

Kullanıcılar, SAIF standart şemasının yeterli olmadığı durumlarda, özellikli sınıf tanımlamaları için de CSN'i kullanırlar. OSN ise transfer veri setinde yer alan sınıf örneklerini tanımlamaya yarar. OSN tanımı doğrudan okunabilir bir tanımlamadır ve CSN'e göre çok daha detaylıdır. Çünkü OSN'de, sınıf hiyerarşisinin bütün dallanmaları, gerçek veri (veri değerleri) ile beraber gösterilir. CSN'de ise, sözkonusu hiyerarşi yalnızca oldukça özet bir şekilde tanımlanır ve gerçek veri bulunmaz. Örneğin, bir CSN Nokta sınıfı tanımından yalnızca, "nokta" nesnelerinin koordinatlarının üç boyutlu olması gerektiği bilgisi alınabilirken, OSN tanımında transfer veri setinde bu sınıf tanımına uyan belli bir nokta nesnesinin gerçek koordinat değerleri görülebilir. CSN ve OSN tanımları birarada bulunabilir ve her ikisi için de ayrıştırıcılar mevcuttur.

Seçmeli Transfer Araçları : SAIF veri modeli seçmeli transferi desteklemektedir. Bir SAIF transfer veri grubu bir ya da daha çok nesne içerebilir. SAIF Toolkit, SAIF çeviricilerinin modüler bir tarzda kolayca geliştirilmesini sağlayacak bir rutin kitaplığı içermektedir. Bu rutinler tek tek nesne bazında işlem yapmaktadır. Ancak, SAIF Toolkit şimdilik uygulama programcılarına yöneliktir. Son kullanıcılara gönderici verisi üzerinde sorgulama ve seçim olanağı sağlayacak bir GKA, [40]'da tanımlanmış olmakla birlikte henüz geliştirilemiştir.

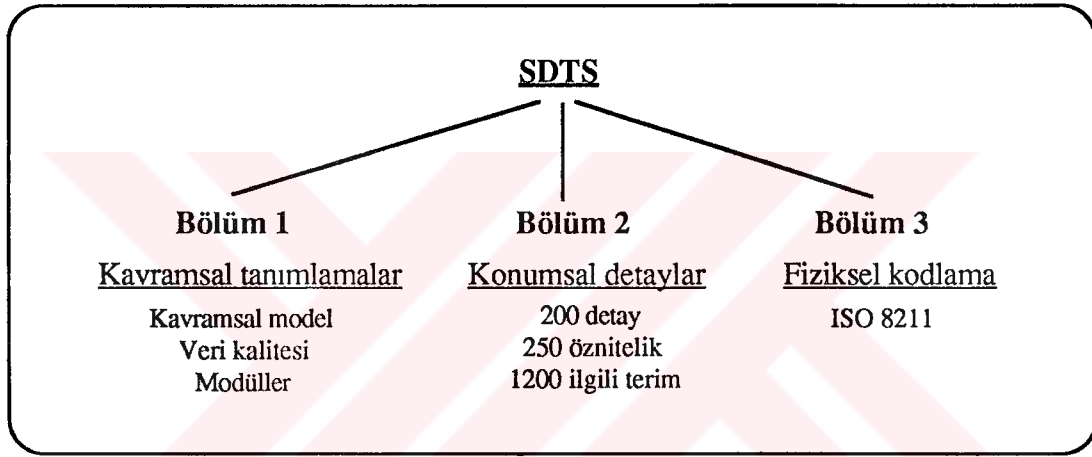
3.2 SDTS (Spatial Data Transfer Standard)

SDTS (Spatial Data Transfer Standard), A.B.D.'de kamu ve özel sektör kuruluşları ile çeşitli üniversitelerden elemanlardan oluşan bir grup tarafından, dokuz yıllık bir sürede geliştirilmiştir [22]. SDTS , Temmuz 1992'de, A.B.D. Ulusal Standartlar ve Teknoloji Enstitüsü (NIST) tarafından federal bilgi işleme standardı (Federal Information Processing Standard, FIPS 173) olarak kabul edilmiştir. Formatın yaygın olarak kabul görmesini sağlamak amacıyla, federal organizasyonların Şubat 1994'den başlayarak SDTS kullanmaları yasal zorunluluk olarak getirilmiştir. SDTS'nin gelişimi ve kullanımını koordine etmek görevi, formatın geliştirilmesinde de liderlik görevini üstlenmiş olan A.B.D. Jeolojik Ölçmeler (USGS) kuruluşuna verilmiştir. Buna göre profil geliştirme, konumsal detay ve öznitelik sözlüğü geliştirme de USGS'nin görevleri arasındadır.

SDTS üç ayrı bölümden oluşmaktadır (Şekil 33). Bir anlamda kavramsal modele ayrılmış olan birinci bölümde, konumsal varlık ve nesneler ile SDTS modülleri ve veri kalitesi kavramları tanımlanmaktadır. Gerçekleştirmede, bir SDTS veri transferi modülleri halinde organize edilir. SDTS'te Genel, Veri Kalitesi, Öznitelik, Konumsal Nesne, Grafik Gösterim gibi temel modüller altında toplam 34 modül tanımlanmıştır [22]. SDTS'in ikinci bölümü ise, konumsal detay tanımlamaya yöneliktir. Bu bölüm 200 topoğrafik ve

hidrografik detay tanımını, 250 öznitelik tanımını, ve 1200 den fazla ilgili terimi içermektedir. Üçüncü kısım fiziksel kodlamayı ele alır. SDTS'te kodlama ISO 8211'e göre yapılır.

USGS daha önce DLG formatında sunduğu 1:100 000 ölçekli harita verisini 1995'te SDTS formatına dönüştürmüştür. USGS, 1996'da da DEM (Digital Elevation Model) ve 1:24 000 ölçekli DLG verilerini SDTS formatına dönüştürmeyi planlamaktadır. Gerek A.B.D.'de SDTS formatında kullanıma sunulan veri miktarı ve gerekse SDTS çeviricilerine sahip olan BDT ve CBS yazılımlarının sayısı henüz oldukça sınırlıdır [135]. Bu bakımdan, SDTS'in 1992'de "standart" olarak onaylandığı da dikkate alındığında, SDTS'in beklenen etkiyi yaratamadığı söylenebilir.

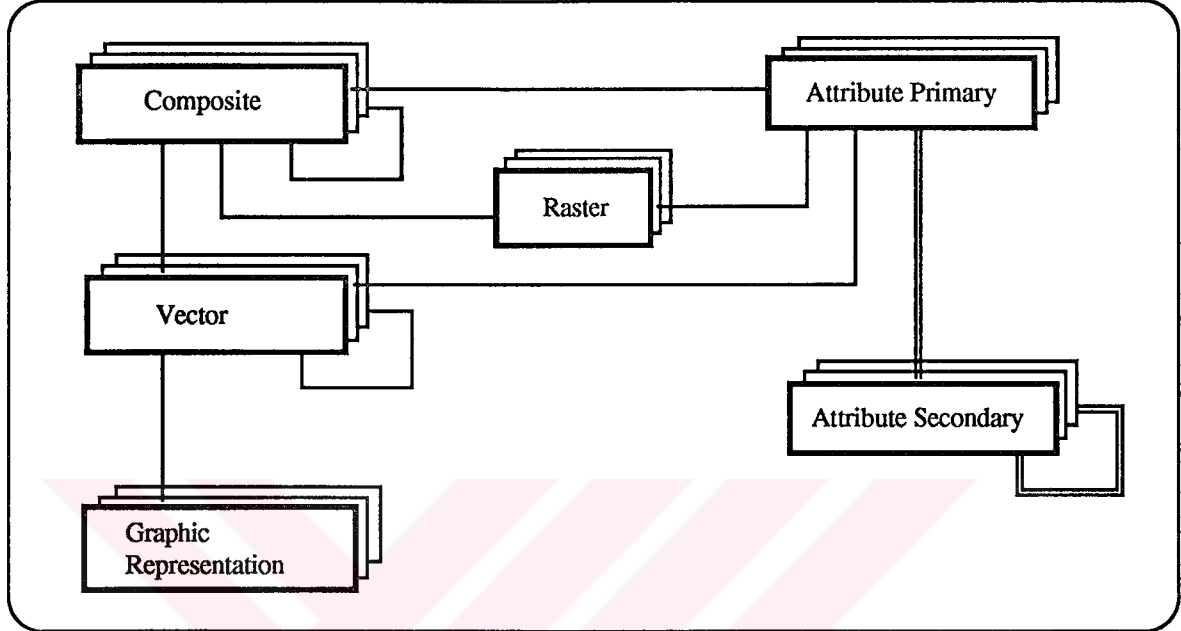


Şekil 33. SDTS'nin organizasyonu.

3.2.1 Veri Modeli

SDTS'te veri modeli ile gerçekleştirim arasındaki ayırım, SAIF'teki kadar belirgin değildir. SDTS, Nesne-Tabanlı fakat, ND olmayan bir veri modeline dayanmaktadır. SDTS kavramsal veri modeli, konumsal varlık (spatial entity) ve konumsal nesne (spatial object) kavramlarını kullanmakta ve konumsal "varlık" "nesne", ve "detay" terimleri arasında bir ayırım yapmaktadır. Buna göre "konumsal varlık" ile, sayısal gösterim konusunu—gerçek dünyayı—oluşturan yol, akarsu, bina gibi varlıklar anlaşılır. SDTS ile ilgili bu kısımda bu terminolojiye bağlı kalınacaktır. Bir konumsal varlığın sayısal bir gösterimi ise bir "konumsal nesne" dir. SDTS sıfır, bir ve iki boyutlu 13 konumsal nesne tanımlamaktadır. Bunlar nokta, çizgi, poligon, pixel gibi, "basit nesneler" olarak ta anılan temel geometrik elemanlardır [22]. "Konumsal detay" ise, konumsal nesne ve konumsal varlık kavramlarının her ikisinin üzerinde, daha genel bir kavramdır. SDTS'te konumsal varlıklar, basit ya karmaşık nesnelerle gösterim edilebilir. Bir karmaşık nesne, basit ve karmaşık nesneler

içerebilir (Şekil 34). Buna göre örneğin "Bölge-karayolu-ağı" karmaşık bir nesne olarak, yine karmaşık nesneler olan "Yerel-karayolu-ağı" nesnelerinden oluşmak üzere tanımlanabilir. Böylece SDTS karmaşık nesne kavramı ile, kullanıcılara özellikli detay tanımlama için belirli derecede bir esneklik sunmaktadır.



Şekil 34. SDTS'nin genel yapısı.

Bir SDTS veri transferi, modüller halinde organize edilir. SDTS'te 5 temel modül içerisinde, toplam 34 modül tanımlanmıştır. SDTS modülleri Şekil 35'te gösterilmiştir. Bir SDTS modülü kayıtlardan, kayıtlar alanlardan, alanlar da alt alanlardan oluşur. Temel SDTS modülleri "Genel", "Veri kalitesi", "Öznitelik", "Konumsal Nesne", ve "Grafik Gösterim" modülleridir. Genel modüller, transfer veri grubuna ait genel bilgilerin (ölçek, referans sistemleri vs.) belirtilmesi ve veri grubunda yer alan konumsal varlık ve özniteliklerin tanımlanması içindir. Konumsal Nesne modülleri vektör, raster, ve karmaşık konumsal nesneler, yani konumsal veri içindir. Veri kalitesi (Data Quality) modülleri veri kalitesinin belirtilmesi içindir. Grafik gösterim (Graphic representation) modülleri, konumsal nesnelerin ve metinlerin kartografik gösterimi içindir.

SDTS'te öznitelik verisi için, "birincil" ve "ikincil" öznitelik modülleri (Attribute primary ve Attribute secondary) bulunmaktadır. Diğer SDTS modülleri kendilerine ait öznitelik verisi için öznitelik modüllerine işaret edebilir. Örneğin konumsal detaylara ait öznitelik verisi için, Öznitelik modülleri konumsal nesne modülleri ile ilişkilendirilir (Şekil 34). Aynı şekilde veri kalitesi kayıtları da veri kalitesi bilgisi içeren öznitelik modül kayıtlarına işaret edebilir. Bu durum Şekil 34'da çizgi bağlantılarla gösterilmiştir. Çift çizgi bağlantılar anahtar öznitelik değeri üzerinden ilişkilendirmeyi, tek çizgi bağlantılar ise anahtar

değerinden bağımsız ilişkilendirmeyi göstermektedir. Her bir öznitelik modülünün yapısı ve özniteliklerin anlamı, domaini, formatı vs. veri sözlüğü modülleri tarafından tanımlanır.

3.2.1.1 Veri Modeli Tipi

SDTS dokümantasyonunda [22] veri modeli "entity-attribute-attribute value" olarak anılmaktadır. SDTS veri modeline göre bir "konumsal fenomen" (spatial phenomena), çeşitli "konumsal detay" (spatial feature) lardan oluşmaktadır. Bu bağlamda SDTS'in nesne-tabanlı bir veri modeline dayandığı söylenebilir [16]. Bununla birlikte SDTS, ND veri modeline dayanmamaktadır. Diğer taraftan, SDTS gerçekleştirim modelinin bütünüyle "ilişkisel" olduğu söylenemez. Çünkü SDTS gerçekleştirimde ilişkisel veri modeli kavramlarını kısmen kullanmakta, örneğin normalizasyon uygulanmamaktadır [136]. Ayrıca, birincil ve ikincil öznitelik tabloları arasındaki bağlantı anahtar öznitelikler üzerinden sağlanırken, birincil öznitelik tabloları ile konumsal nesne modülleri arasındaki bağlantı öznitelik değerlerinden bağımsız kayıt numaralarıyla sağlanmaktadır.

3.2.1.2 Raster Veri

Şekil 34 te görüldüğü gibi, SDTS'te raster veri için "Raster Definition" ve "Cell" Modülleri bulunmaktadır. "Raster Definition" kodlama formatını tanımlar. Cell modülü ise göze (pixel) verisini içerir. SDTS, şimdilik en çok kullanılan dört raster formatını tanımaktadır. Bunlar "Layer sequential", "Layer interleaved by line", "Layer sequential run encoding", ve "Run encoding with attributes for each cell". Cell modül kayıtları, öznitelik kayıtlarına bağlanabilir. Böylelikle, öznitelik değerlerinin gözelere atanmasında esneklik sağlanmış olur.

3.2.1.3 Kartoğrafik Gösterim

SDTS, kartoğrafik gösterim bakımından oldukça güçlüdür. SDTS bunun için, "Graphic Representation" modüllerini içermektedir (Şekil 34). "Text Representation" modülü yardımıyla, metinler oldukça ayrıntılı olarak pozisyon, yöneltme, karakter büyüklüğü ve genişliliği, renk, font ve diğer karakteristikleri ile belirtilebilir. Böylelikle eğri çizgiyi izleyen metinler belirtilebilmektedir. "Symbol Representation" modülü ile nokta, çizgi, ve poligon işaretleri büyüklük, yöneltme ve renkleri ile belirtilebilir. "Color Index" modülü text ve işaret renklerinin, "Font Index" modülü ise, metin fontlarının belirtilmesi içindir. Böylece istenen renk ve font ta grafik/metin gösterimi için, Graphic Representation modülleri, Color ve Font Index modülleri ile ilişkilendirilir.

Projeksiyonlar açısından, SDTS'te orijinal olarak yalnızca üç projeksiyon sistemi tanımlanmıştır. Bununla birlikte, kullanıcılar kendi projeksiyon sistemlerini tanımlayabilir (bkz. 3.2.1.4).

3.2.1.4 Koordinatlar

SDTS'de üç boyutlu detaylar doğrudan tanımlanamamaktadır. Diğer bir anlatımla SDTS veri modelinde konumsal nesneler için üçüncü boyut belirtilmemiştir [136]. Bununla birlikte karmaşık nesneler kullanılarak koordinatlar üç boyutlu olarak transfer edilebilir [22], [121]. Benzer şekilde üç boyutlu detaylar daha düşük boyutlu detaylardan oluşan karmaşık nesneler olarak tanımlanarak transfer edilebilir. Koordinatlar mutlak ve bağıl koordinatlar olarak kodlanabilir. SDTS zaman boyutunun bir koordinat değeri olarak kodlanması için herhangi bir destek sunmamaktadır.

Koordinat sistemlerinin belirtilebilmesi için SDTS'te "Spatial Reference" modülü ve onun altında dört ayrı modül tanımlanmıştır (Şekil 34). "Internal Spatial Reference" modülü, koordinat formatlarının belirtilmesi içindir. "External Spatial Reference" modülü ise, referans sistemleri ve projeksiyon sistemleri ile birlikte yer koordinat sistemini tanımlar. Standart olarak desteklenen sistemler coğrafi koordinat sistemi, UTM, ve Eyalet düzlem koordinat sistemi (SPCS) dir. Bununla birlikte, kullanıcılar kendi koordinat sistemlerini belirtebilirler. Bunun için, External Spatial Reference modülü özellikli koordinat sistemi tanımını içeren "Attribute Primary" modülü ile ilişkilendirilir. Spatial Reference modülü vasıtasıyla, aynı bir transfer veri grubu içerisinde farklı koordinat sistemleri belirtilebilir [22]. Koordinatların kodlanmasında, SDTS pratik olarak sınırsız bir çözünürlük sunmaktadır.

3.2.1.5 Topoloji-Geometri Ayırımı

SDTS'te topoloji-geometri ayırımı özellikle vurgulanmakta ve yalnızca geometri ya da geometri ve topolojiyi birlikte içeren konumsal nesneler ayrı sınıflar olarak tanımlanmaktadır. Yalnızca geometri içeren nesneler "G" ile, hem geometri hem de topoloji içeren nesneler ise "GT" ile işaretize edilir. Örneğin, SDTS'te "Line Segment" ve "Link" farklı konumsal nesneler olarak tanımlanmaktadır. Geometrik olarak her iki tip nesne de iki noktadan oluşmakla birlikte, Link nesnesi, yalnızca geometri içeren Line Segment nesnesine ek olarak, iki nokta arasındaki bağlantının yönünü (hangi noktadan hangi noktaya olduğunu) de içermektedir. Benzer şekilde 0-boyutlu nesneler için "Point" ve "Node", 2-boyutlu nesneler için "G-Polygon" ve "GT-Polygon" nesneleri bulunmaktadır [22].

3.2.1.6 Karmaşık Nesneler

SDTS' de tanımlı konumsal nesne tiplerinden biri de karmaşık nesneleri tanımlamaya yarayan "Composite" dir. Bir karmaşık (composite) nesne basit raster, vektör nesnelerini ya da yine karmaşık nesneleri içerebilir (Şekil 34). SDTS'te bir birincil modül kaydı (detay kaydı), en çok bir konumsal nesne ile eşleştirilebilir. Bununla birlikte her bir konumsal nesne, birden çok detay tarafından işaret edilebilir [22], [136]. Bu durumda, karmaşık bir detayın oluşturulabilmesi için, önce onu gösterim edecek karmaşık konumsal nesnenin oluşturulması gerekir. Dolayısıyla SDTS'te karmaşık detaylar, ancak konumsal nesne yani geometri düzeyinde tanımlanabilir. Karmaşık detay öznitelikleri, aynen basit nesnelerde olduğu gibi öznitelik modülleri yardımıyla gösterim edilir. SDTS tasarım felsefesi, karmaşık nesne yapısı ile herhangi bir özellikli detay tanımlamasının yapılabilmesini amaçlamıştır. Ancak "karmaşık detay" kavramı, konumsal nesne düzeyinde olması bir yana, oldukça genel olarak tanımlanmış, amaca uygun karmaşık konumsal nesnelerin tanımlanması gerçekleştirimcilere bırakılmıştır.

3.2.1.7 Nesne Paylaşımı

SDTS'te konumsal varlıklar genel olarak birincil öznitelik, ayrıntılı olarak ta ikincil öznitelik modüllerinde tanımlanır. Buna göre her bir konumsal varlık için bir birincil öznitelik modülü kaydı ve bir ya da daha çok ikincil öznitelik kaydı bulunur. SDTS'te bir konumsal nesne kaydı, birden çok birincil öznitelik kaydı ile ilişkilendirilebilir. Diğer bir anlatımla, birden çok konumsal varlık aynı konumsal nesne ile gösterim edilebilir. Örneğin bir göl ve arazi parseli aynı çizgi parçasına (SDTS'te "link") işaret edebilir. Dolayısıyla, SDTS nesne paylaşımını konumsal nesneler (geometri) düzeyinde desteklemektedir. Bunun sakıncası, bir karmaşık varlığa ait konumsal olmayan öznitelik bilgisinin elde edilmesinin, tamamiyle nesne-tabanlı gerçekleştirimlere göre—SAIF gibi—ayrı bir arama gerektirmesidir.

3.2.1.8 Meta Veri

SDTS'te meta veri için "Data Quality" modülleri bulunmaktadır (Şekil 35). Her bir veri transferi için bu modüllerin belirtilmesi zorunludur. Bunlardan "Lineage" modülü, verinin işlemsel geçmişini yani hangi kaynaklardan nasıl derlendiğini belirtmek içindir. "Positional accuracy" ve "Attribute accuracy" modülleri ise, sırasıyla konumsal ve konumsal olmayan verinin doğruluğunun belirtilmesi içindir. "Logical consistency" modülü ile konumsal veri tutarlılığı belirtilir. "Completeness" modülü, transfer veri grubunun ait olduğu coğrafi bölge ve varsa transfer veri grubunu belirleyen seçim ölçütleri gibi ayrıntılar içerir.

SDTS'te meta veri, transfer veri grubunun tamamı, belli bir alt grubu (değişik düzeylerde; katman, konumsal kapsam vs.), ya da tek tek detaylar düzeyinde belirtilebilir. Buna bağlı olarak bir veri kalitesi modülü kayıtlı çeşitli formlar alabilir. Buna göre veri kalitesi kayıtlı (1) metinsel olabilir (veri kalitesi raporu şeklinde), (2) veri kalitesini tanımlayan öznitelik kayıtlarına işaret edebilir ve/veya (3) başlıbaşına diğer modül kayıtlarına ilişkin veri kalitesi bilgilerini içerebilir [22].

3.2.1.9 Nesne Bazında Veri Değişimi

SDTS ile nesne bazında veri değişimi uygulanabilir. Ancak, önemli olan bunun ne kadar pratik olacağıdır. SDTS esasen, dosya bazında veri değişimine yönelik olarak tasarlanmıştır [15], [137], [138]. Bu nedenle, mevcut SDTS transfer veri grubu yapısı ile nesne bazında veri transferi pek verimli olmayacaktır. Örneğin TVP de, her bir SDTS modülüne ait verinin ayrı bir dosyada tutulması gerekir. Tek bir detay için bir çok dosyanın oluşturulması ve transferi pratik değildir. Dolayısıyla, SDTS gerçekleştirim modeli, nesne bazında veri transferi için uygun değildir.

3.2.1.10 Zamansal Veri

SDTS'de zamansal veri desteği bulunmamaktadır.

3.2.1.11 Multimedya

SDTS, multimedya desteği sunmamaktadır.

3.2.1.12 Genişletilebilirlik

SDTS karmaşık nesne kavramını desteklediği için, herhangi bir nesne gerçekleştirimi zor olmakla birlikte (bkz. 3.2.1.6) gösterim edilebilir. Ayrıca, SDTS'in detay ve özniteliklerin kodlanması için kullandığı kendine ait katalogu genişletilebilmekte, yeni detay tanımları eklenebilmektedir. Ancak, SDTS'in mevcut veri modelinin tanımladığı veri tiplerinin yeterli olmaması durumunda, yeni veri tiplerinin eklenmesi ya da mevcut veri tiplerinin yeniden tanımlanmasına yönelik olarak kullanıcılara bir destek sunmamaktadır. Bu tür genişletmeler ancak belirli bir otorite—USGS gibi—tarafından yapılabilir. SDTS'in detay kodlama katalogu açısından da durum aynıdır; kataloga ekleme ya da düzeltmeler, farklı kullanıcı gereksinimleri doğrultusunda yine belirli bir otorite tarafından yapılabilir. Özetle SDTS, genişletilebilir bir veri modeline sahip olmaması ve karmaşık nesnelerin gösteriminde 3.2.1.6 da anılan güçlük nedeniyle ancak sınırlı düzeyde bir genişletilebilirlik sunabilir.

3.2.2 Gerçekleştirim

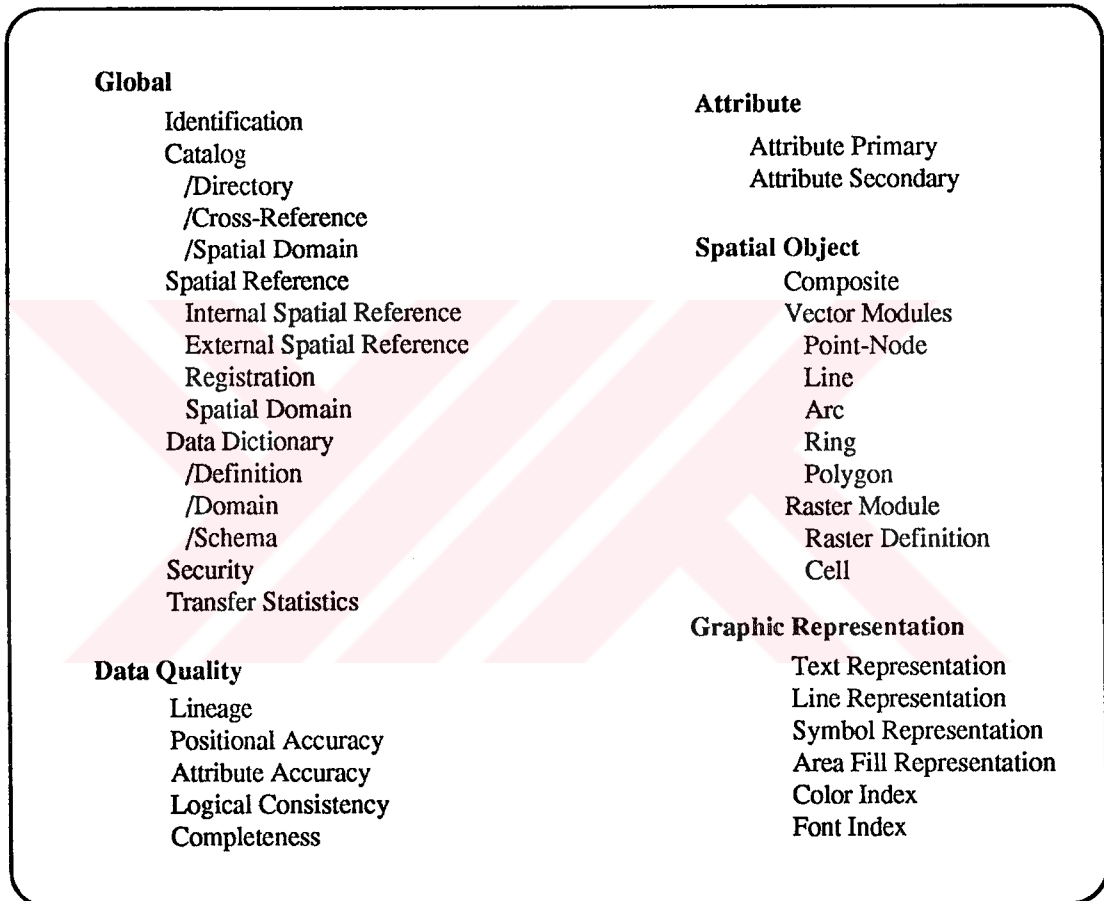
Bir SDTS veri transferi, modüller halinde organize edilir. SDTS'te 5 temel modül içerisinde, toplam 34 modül tanımlanmıştır. SDTS modülleri şekil 35'de gösterilmiştir (Şekilde modül isimleri, pratik bir yararı olmayacağı düşüncesiyle, türkçeye çevirilmemiştir). "/" ile başlayan modüller, seçmeli modüllerdir. Modül oluşum hiyerarşisi modül_kayıt_alan_alt alan şeklindedir [22]. Bu bakımdan SDTS modülleri, ilişkisel veri modeli ilişkilerine (tablolar) benzetilebilir. Ancak bu ilişkiler, ilişkisel veri modelinde olduğu gibi normalize edilmiş ilişkiler değildir. SDTS'te birincil ve ikincil öznitelik tabloları arasındaki bağlantı anahtar öznitelikler üzerinden sağlanırken, birincil öznitelik tabloları ile konumsal nesne modülleri arasındaki bağlantı öznitelik değerlerinden bağımsız kayıt numaralarıyla sağlanmaktadır. Dolayısıyla gerçekleştirimde kodlayıcı, bu tek anlamlı kayıt numaralarını üretmek durumundadır.

Bir veri grubunun SDTS formatında kodlanması, kavramsal düzeyden fiziksel düzeye bir dizi dönüşüm içerir. Bu dönüşümler sırasıyla kavramsal dönüşüm, modül dönüşümü, ve format dönüşümüdür [121]. Kavramsal dönüşüm, veri grubunun kavramsal şeması ile SDTS kavramsal şeması arasındaki dönüşümü tanımlar. Bunun için öncelikle uygun bir SDTS profili belirlenir. Modül dönüşümü, veri grubu veri elemanlarına karşılık gelen modüllerin ve modül oluşum hiyerarşilerinin ve modüller arası ilişkilerin belirlenmesini içerir. Bu aşamada ayrıca, SDTS kapsamında bulunmayan özellikli detay/öznitelikler için gerekli öznitelik modülleri tanımlanır. Bu bakımdan modül dönüşümü, alışlagelmiş veri tabanı tasarımındaki kavramsal şema ile gerçekleştirim şeması arasındaki dönüşüme benzetilebilir. Belirli bir veri transferinde 34 SDTS modülünün her biri yer almayabilir. Gerekli modül tipleri kavramsal dönüşüm aşamasında uygun profilin belirtilmesi ile zaten büyük ölçüde belirlenmiş olur. Format dönüşümü, modül dönüşümü sonucunda belirlenen modüller ile ISO 8211 arasındaki dönüşümü ve verinin ISO 8211 dosyalarına aktarılmasını içerir. Bunun için ISO 8211 dosya, kayıt, ve alan yapıları ve transfer ortamı belirlenmelidir.

3.2.2.1 Profiller

SAIF'te olduğu gibi, SDTS tasarım felsefesi de formatın profiller kullanılarak gerçekleştirimini öngörmektedir [139]. Bunun nedeni, her iki formatın da oldukça genel olmasıdır. Gerçekleştirimde öncelikle uygun bir profilin belirlenmesi, kavramsal dönüşüm, modül, ve format dönüşümü aşamalarının yükünü önemli ölçüde hafifletir [121]. SDTS için öngörülmüş olan profillerden biri, SDTS TVP profili (Topological Vector Profile) dir. SDTS TVP, geometri ve topoloji içeren vektör veriye yöneliktir. SDTS TVP'de Graphic Representation modülü bulunmamaktadır. SDTS'in 4. bölümü olarak kabul edilen SDTS

TVP'nin 1.0 versiyonu, Haziran 1994'de kullanıma sunulmuştur. Şu an mevcut ve geliştirilmekte olan SDTS çeviricileri, SDTS TVP'ye yönelik çeviricilerdir [137]. Raster veriye yönelik olarak geliştirilmiş olan Raster profili teknik olarak tamamlanmış olup, resmi onay için gözden geçirilmektedir [137]. Ne zaman kesinlik kazanacağı henüz belli olmayan Raster profilinin de, SDTS'in 5.'inci bölümünü oluşturması düşünülmektedir. Ayrıca, son yıllarda önemli bir ivme kazanmış olan, farklı konumsal veri değişim formatları arasında uyum sağlanması çabaları çerçevesinde geliştirilen DIGEST ve DX-90 formatlarına yönelik profiller, sırasıyla 1993 ve 1994'de tamamlanmıştır [23].



Şekil 35. SDTS modülleri.

3.2.2.2 Detay Kodlama

SDTS'in ikinci bölümü, konumsal detay tanımlamaya yöneliktir. Bu bölüm 200 topoğrafik ve hidrografik detay tanımını, 250 öznitelik tanımını, ve 1200 den fazla ilgili terimi içermektedir. Bunun yanında SDTS Özellikli detay tanımlamaları için veri sözlüğü kullanılmasını desteklemektedir. Diğer bir anlatımla herhangi bir transfer için SDTS'te standart olarak tanımlı detay/özniteliklerin yeterli olmaması durumunda kullanıcılar kendi tanımlamalarını bir veri sözlüğü yardımıyla ekleyebilirler. Özellikli detay tanımlamalarının

veri sözlüğü vasıtasıyla transferi için "Data Dictionary" modüllerinin transfer veri grubunda yer alması gerekir. "Data Dictionary" modülleri ile, detay/öznitelik tanımlamaları, öznitelik domainleri, ve bunlarla ilgili öznitelik modül yapıları belirtilir. Transfer veri grubuna ait bir veri sözlüğünün varlığı ayrıca, "Identification" modülü içerisinde belirtilir [22]. SDTS ile, SDTS'nin kendi "iç" katalogundan başka, "dış" kataloglar ya da detay kodları kullanılabilir [137].

3.2.2.3 Kodlama Yöntemi

SDTS'te kodlama için genel amaçlı bir veri değişim standardı olan ISO 8211 [78] kullanılır. SDTS modüllerinin ISO 8211 dosyalarına aktarılması, SDTS'in 3. bölümünü oluşturmaktadır. SDTS'te veri modeli ve kodlama yöntemi birbirinden bağımsız olarak ele alındığı için, herhangi bir kodlama standardı kullanılabilir [22]. Bununla birlikte, şimdiye kadar hep ISO 8211 kullanılmıştır.

Bir ISO 8211 dosyası veri ile birlikte, veri tanımını da içermektedir. ISO 8211'de etiketli kodlama uygulanır. Etiketlerin anlam ve yapıları da dosya içerisinde açıklandığı için, kod çözümü doğrudan dosya üzerinden yapılabilir. Bu nedenle, ISO 8211 dosyaları çoğunlukla "Veri Tanımlayıcı Dosya" olarak anılır. Bununla birlikte veri tanımlamanın oldukça ayrıntılı olarak yapılması transfer veri grubunun boyutunu büyütür. ISO 8211'de kodlama ASCII ya da ikili modda yapılabilir [78].

ISO 8211, kökleri 1970 lere uzanan bir standart olarak çevirim dışı veri transferine yöneliktir. Dolayısıyla, desteklenen transfer ortamları manyetik bant ve kartuşlar, disketler, ve kompakt diskler (CD-ROM) dir. ISO 8211'in iletişim ağları desteği yoktur. Bu nedenle veri transferi ancak, dosya transferi şeklinde gerçekleştirilebilir. SDTS modüllerinin ISO 8211 dosyalarına aktarılmasında, SDTS bazı sınırlamalar koymuştur. Buna göre, her bir "Global" modül için bir ISO 8211 dosyası bulunur. Diğer modüller için böyle bir sınırlama yoktur. SDTS TVP'deki sınırlama daha katıdır ve her bir modül için ayrı bir dosya gerektirir. Günümüz ihtiyaçları karşısında kaçınılmaz hale gelen "Federe CBS ler", transfer verisinin iletişim ağları üzerinden "veri dizisi" olarak transfer edilebilmesi gerektirmektedir. Bu bakımdan, ISO 8211'in iletişim ağları desteğinin olmaması çok önemli bir eksiklikler.

3.2.3 Değişim Ortamı

SDTS veri modeli servisleri desteklememektedir. SDTS için önce, "FIPS 123 rutin kitaplığı" geliştirilmiştir. "C" dilindeki bu kitaplık, SDTS formatındaki verinin ISO 8211'e kodlanması ve tersi yönde ISO 8211 verisinin SDTS formatına dönüştürülmesi için gerekli

kodlama/kod çözümleme rutinlerini kapsamaktadır. SDTS çevirici geliştirilicilerine yönelik daha geniş kapsamlı bir destek için ayrıca, "CSP" (Common Software Platform) "toolkit"i geliştirilmektedir. C++ dilinde yazılmış olan CPS henüz prototip aşamasındadır. SDTS seçmeli transferi desteklememektedir.

3.2.3.1 Dokümantasyon

SDTS oldukça karmaşık bir format olarak kabul edilmekte, formatın anlaşılması ve gerçekleştirimin oldukça güç olduğu çeşitli platformlarda dile getirilmektedir [140], [141]. Bunda SDTS'nin çok kapsamlı bir format olması yanında, dokümantasyonun kolayca anlaşılabilir olmamasının rolü büyüktür. Dokümantasyon yetersizliğinin başlıca iki nedeni vardır. Birincisi, veri modeli ve gerçekleştirim kavramlarının çoğunlukla birbirine karıştırılmış olmasıdır. Örneğin, SDTS dokümantasyonu veri modellerine veya ND modele herhangi bir göndermede bulunmamaktadır. İkinci neden, standart olmasa da bugün yaygın olarak kullanılan bir terminolojinin kullanılmaması, onun yerine yeni birtakım terimlerin kullanılmasıdır. Ayrıca göndermede bulunulduğu halde hiç açıklanmayan kavramlar vardır. Standart olarak önerilen bir değişim formatının, standart olmayan bir terminoloji ile kullanıma sunulması, formatın anlaşılması ve gerçekleştirimini güçleştirecek, dolayısıyla geniş bir kullanım alanı bulmasını olumsuz yönde etkileyecektir. [142], standartlaştırmada terminolojinin önemini vurgulamaktadır.

3.2.3.2 Servisler

SDTS için "FIPS 123 rutin kitaplığı" geliştirilmiştir. Bu kitaplık C dilinde yazılmış olup 30 civarında rutin içermektedir. Bu rutinler SDTS formatındaki verinin ISO 8211'e kodlanması ve tersi yönde ISO 8211 verisinin kod çözümlemeye geçirilerek SDTS formatına dönüştürülmesi için gerekli kodlama/kod çözümleme rutinleridir [139]. SDTS çevirici geliştirilicilerine yönelik daha geniş kapsamlı bir destek için ayrıca, "CSP" (Common Software Platform) "toolkit"i geliştirilmektedir. C++ dilinde çevirici modülerini içeren CPS, henüz prototip aşamasındadır [137].

3.2.3.3 Değişim Ortamı Araçları

SDTS'nin değişim ortamı araçları desteği çok zayıftır. Derleyici araçları bulunmamaktadır. VTD olarak BNF kullanılabilir. Seçmeli transfer araçları bulunmamaktadır.

3.3 DIGEST (Digital Geographic Exchange Standard)

DIGEST, NATO'nun 1983'te oluşturduğu bir çalışma grubu (DGIWG, Digital Geographic Information Working Group) tarafından geliştirilmiştir. Bu grubun onbir üyesi bulunmaktadır. Yakın zamanda Avusturalya ve Yeni Zelanda gözlemci statüsüyle katılmıştır. Türkiye DGIWG'de yer almamıştır. DIGEST'in ilk versiyonu, Haziran 1991 tarihli'dir. Ekim 1992'de DIGEST 1.1 versiyonu yayımlanmıştır. Aralık 1993'teki bir dizi değişikliğin ardından, 1994 başlarında DIGEST 1.2 kullanıma sunulmuştur. Ocak 1993'te NATO, DIGEST'i kendi veri değişim standardı olarak kabul etmiştir [124].

DIGEST, üreticiden kullanıcıya ve kullanıcıdan kullanıcıya veri değişimine yönelik olarak tasarlanmıştır. Üreticiden kullanıcıya etkin veri transferi ancak, alıcının transfer verisini tercihen grafik bir görüntü ekranında görüntüleyip, sorgulayarak transfer edebilmesiyle mümkün olacaktır. Bu noktadan hareketle kullanıma sunulan DCW (Digital Chart of the World) ürünü, 1:1 000 000 ölçeğinde, Dünya genelinde temel harita verisi yanında söz konusu fonksiyonları sunmaktadır. DCW, DIGEST'in popülerite kazanması ve kullanımının yaygınlaşmasında çok önemli bir rol oynamaktadır.

DIGEST, SAIF ve SDTS'e göre daha dar kapsamlı bir konumsal veri değişim formatı olarak kabul edilmektedir. Bu nedenle DIGEST bazen "özel" (defined), SAIF ve SDTS te "genel" standartlar olarak anılır [23], [143]. Buna rağmen DIGEST'in değerlendirmeye alınmasının nedenlerinden biri, son yıllarda adından en çok söz ettiren ve gelecek vadeden bir format olmasıdır. Diğer neden ise, bir NATO üyesi olan Türkiye'nin DIGEST'e kayıtsız kalamayacağı ve zaten bu açıdan bir takım gelişmelerin yaşanıyor olmasıdır (bkz. bölüm 1).

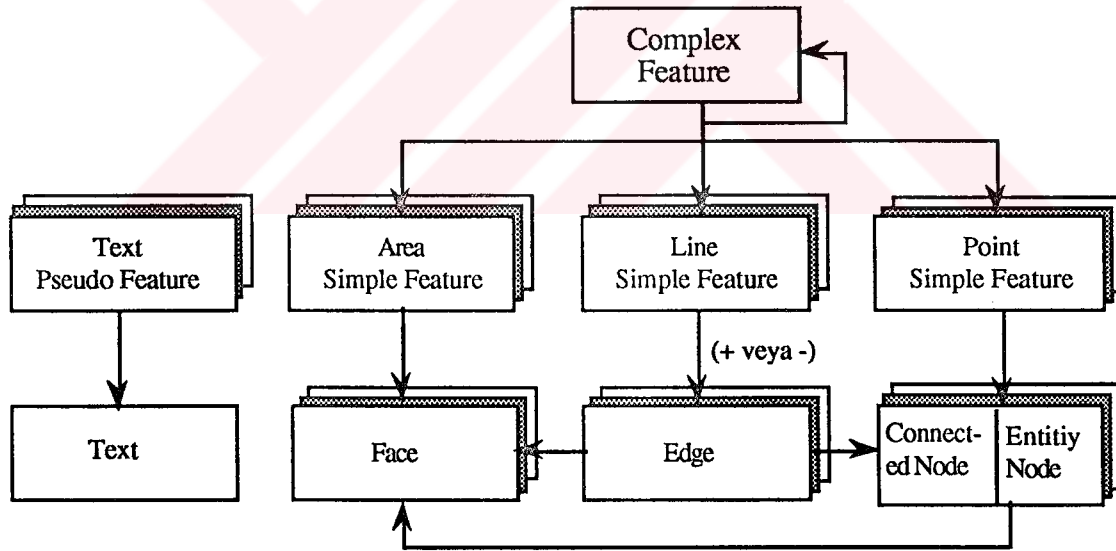
3.3.1 Veri Modeli

DIGEST, özellikle dört tip veriye yöneliktir. Bu veri tipleri, DIGEST terminolojisi ile, topolojik vektör veri, spagetti vektör veri, raster veri ve matris veridir. DIGEST'te bu tip veri, nesne-tabanlı ya da kayıt-tabanlı bir tarzda gösterim edilebilir. Diğer bir ifadeyle, DIGEST ile veri değişimi nesne-tabanlı veya kayıt-tabanlı tarzda gerçekleştirilebilir. Dolayısıyla, DIGEST hem nesne-tabanlı hem de kayıt-tabanlı veri modeli kavramlarını desteklemektedir. DIGEST'in nesne-tabanlı gerçekleştirimi DIGEST-A, kayıt-tabanlı gerçekleştirimi ise DIGEST-C olarak anılır [16]. DGIWG DIGEST-C ile ilgili çalışmalarına 1989'da başlamıştır.

DIGEST-A'da detaylar basit ve karmaşık detaylar olarak tanımlanmaktadır. Her bir detay geometrisi, öznitelikleri ve diğer detaylarla olan ilişkileri ile tanımlıdır. Karmaşık

detaylar basit ya da yine karmaşık detaylardan oluşur. DIGEST veri modeli Şekil 36'de görülmektedir. Şekil 36'de ok yönleri, detay oluşumuna işaret etmektedir. Buna göre karmaşık detaylar (complex features) alansal (area), çizgisel (line), ve noktasal (point) basit detaylar (simple features) ya da karmaşık detaylar içerebilir. Benzer şekilde alansal, çizgisel, ve noktasal basit detaylar da yüz (face), kenar (edge), düğüm (node) gibi topolojik (geometrik) varlıklar içermektedir. Topolojik varlıklar detayların geometrisini—koordinatlarla— ve birbirleri ile olan konumsal ilişkilerini gösterim ederler. Topolojik varlıklar ve detaylar arasındaki bu oluşum ilişkisi genelde tek yönlü olmakla birlikte, gerçekleştirimde işlemsel verimliliği artırmak amacıyla çift yönlü olarak belirtilebilir [144].

DIGEST-A ve DIGEST-C kapsamındaki veri tipleri aynıdır. Ancak DIGEST-C ilişkisel veri modeline dayanmaktadır. Dolayısıyla veri, ilişkiler kullanılarak organize edilir. Bu durumda bir detayın çeşitli tablolarındaki verilerden oluştuğu, ilgili anahtar alanlar yardımıyla belirtilir. Topolojik varlıklarla detaylar arasında Şekil 36'de görülen ilişkiler burada çift yönlü olabilir. Bu yolla bire-bir, bire-çok, ve çok-a-çok ilişkiler gösterim edilebilir. İlişkisel işlemler kullanılarak veriye grup işlemleri uygulanabilir. Örneğin ilişkisel birleştirme yoluyla basit detaylar üzerine karmaşık detaylar tanımlanabilir.



Şekil 36. DIGEST veri modeli.

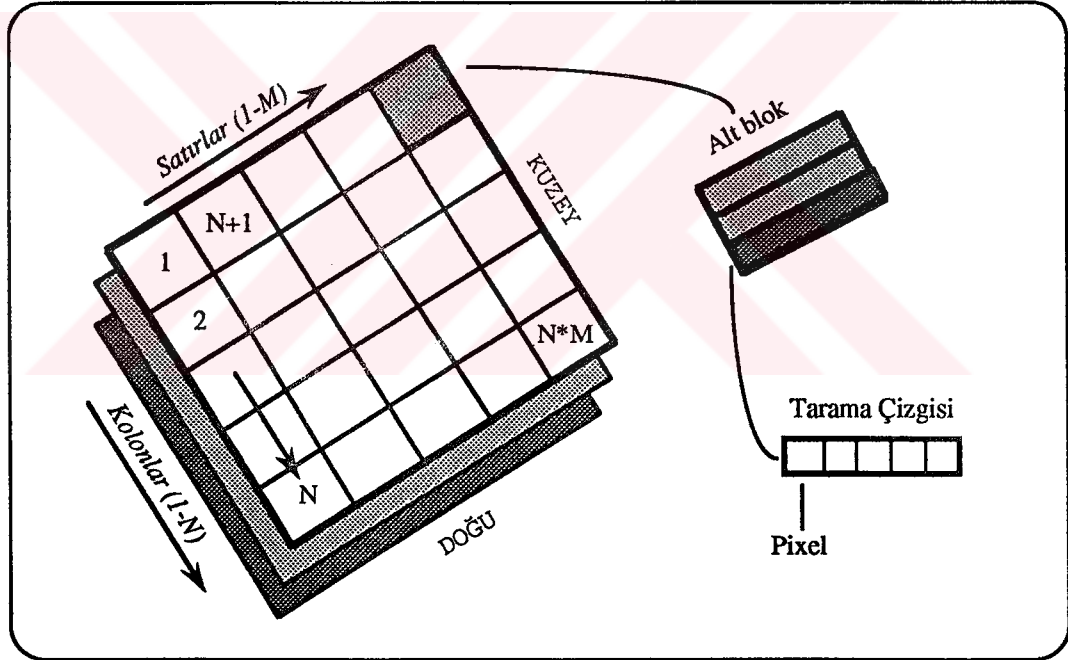
3.3.1.1 Veri Modeli Tipi

DIGEST ile veri değişimi nesne-tabanlı veya kayıt-tabanlı tarzda gerçekleştirilebilir. Dolayısıyla, DIGEST hem nesne-tabanlı hem de kayıt-tabanlı veri modeli kavramlarını desteklemektedir. DIGEST'in nesne-tabanlı gerçekleştirimi DIGEST-A, kayıt-tabanlı

gerçekleştirimi ise DIGEST-C olarak anılır [16]. DIGEST-A değişken uzunluklu detay kayıtlarını kullanır. Bir detay kayıtlı diğer detay kayıtlarına göstergeçler bulundurabilir ve öznitelik verisini içerir. DIGEST-C ise, ilişkisel veri modelini kullanmaktadır.

3.3.1.2 Raster Veri

DIGEST'de raster veri düzenli gridlerle gösterim edilir. Gridler "alt blok" lardan, alt bloklar da pixellerden oluşur. Buna göre, kuzey-güney doğrultusunda N satır, ve doğu-batı doğrultusunda M sütun bulunan bir grid, $N \times M$ alt blok içerir (Şekil 37). Alt bloklamanın espris, veriye ulaşım zamanını azaltmaktır. Grid içerisinde alt blokların, ve alt blok içerisinde pixellerin numaralandırılmasında orijin ve artırım yönü seçimi için çeşitli seçenekler mevcuttur. Kartezyen koordinat sistemi ile uyumlu bir seçimde, orijin olarak güneybatı köşesi alınır ve artırımlar batıdan doğuya ve güneyden kuzeye doğru uygulanır. Söz konusu seçenekler farklı formatlarda raster veri transferine olanak tanımaktadır.



Şekil 37. DIGEST raster veri modeli.

Bir önceki bölümde belirtildiği gibi raster veri gösteriminde aynı bir konumsal kapsam için birden çok karakteristiğin gösterimi sözkonusu olduğunda, aynı bir grid bölümlemesine birden çok karakteristik değeri karşılık gelecektir. Karakteristikler yükseklik, toprak cinsi vb., ya da hava fotoğrafları veya uydu görüntülerine ait radyometrik yansıma değerleri olabilir. DIGEST, radyometrik değerleri "raster" veri, diğer tip karakteristikleri de "matrix" veri olarak birbirinden ayırmakta, ancak her iki tip için de, yukarıda açıklanan aynı veri yapısını kullanmaktadır [144]. O nedenle, yazara göre sözkonusu ayırım gereksizdir.

3.3.1.3 Kartografik gösterim

DIGEST'in kartografik gösterim gücü SAIF ve SDTS'e göre çok daha azdır. Çünkü DIGEST'te yalnızca isim ya da metinler belirtilebilmekte, diğer işaretler için bir destek sunulmamaktadır. Bir haritada metinler bir detaya ait olabilir ya da kendi başına bir metin olabilir. Bu nedenle DIGEST metinleri de birer detay (pseudo feature) olarak görür (Şekil 36). Bir metin için, varsa metnin ait olduğu detay yanında, metnin konumu ve diğer öznitelikleri belirtilebilir. Öznitelikler eldeki transfere uyarlanabilir [144].

DIGEST'in projeksiyon sistemi desteği iyidir. Projeksiyonlar iki karakterlik kodlarla ve dört parametre ile karakterize edilir. Bu dört parametre her projeksiyon için kullanılmayabilir. Bu da projeksiyon kodundan belirlenebilir.

3.3.1.4 Koordinatlar

DIGEST'te koordinatlar ancak 2 ya da 3 boyutlu olarak belirtilebilir. Projeksiyon ve referans sistemi desteği açısından, DIGEST oldukça güçlüdür. Bunda, formatın çok uluslu bir çalışma grubu tarafından hazırlanmış olmasının rolü büyüktür. DIGEST mutlak, bağıl, ve diferansiyel koordinat kodlama desteği sunmaktadır. Koordinat çözünürlüğü pratik olarak sınırsızdır. Bu özellikleriyle DIGEST'in MACDIF [80]'ten önemli ölçüde etkilendiği gözlenmektedir.

3.3.1.5 Topoloji-Geometri Ayırımı

DIGEST, topoloji-geometri ayırımını desteklemektedir. DIGEST topolojik verinin belirtilmesinde dört farklı düzey sunmaktadır. 0. düzeyden başlayarak bu düzeylere karşılık gelen veri tipleri sırasıyla spagetti vektör, chain-node vektör, planar graph vektör ve full topolojik vektör olarak anılmaktadır. Diğer bir anlatımla spagetti vektör veri 0, full topolojik vektör veri de 3. düzey topoloji içermektedir. Dolayısıyla topolojik veri gerektirmeyen durumlarda spagetti veri yapısı kullanılabilir. Farklı düzeylerdeki topolojik veri gereksinimi ise ilgili düzeye ait veri yapısı kullanılarak sağlanır.

Spagetti veri yapısı ile ancak nokta ve çizgi detaylar gösterim edilebilir. Alanlar ise alanı çevreleyen "kapalı-çizgi"ler yardımıyla gösterim edilebilir. Spagetti veri yapısında "düğüm" kavramı—çizgilerin kesişme noktaları—söz konusu olmadığından nesne paylaşımı uygulanamaz. Bu durumda örneğin iki parsel arasındaki ortak sınır verisi, her iki parsel için de ayrıca kodlanmalıdır. Nesne paylaşımının uygulanabilmesi için chain-node veri yapısının kullanılması gerekir. Chain-node veri yapısında her çizgi parçasının bir başlangıç ve bitiş

düğümü vardır. Başlangıç ve bitiş, sayısallaştırma yönü ya da çizgi parçasının koordinat listesinde koordinatların sırasına göreler.

3.3.1.6 Karmaşık Nesneler

Yukarıda da belirtildiği gibi DIGEST-A kullanılarak karmaşık detaylar basit ya da karmaşık detaylar üzerine tanımlanabilir. Karmaşık detay tanımlamada özyineleme sınırlaması yoktur. DIGEST-C de karmaşık detay tanımlamaya olanak tanımakla birlikte, karmaşık detay aynı bir katman içinde olmalıdır. Diğer bir ifadeyle, birden çok katmanla ilgili karmaşık detaylar tanımlanamamaktadır. DIGEST-C de karmaşık detaylar iki şekilde tanımlanabilir. Bir yaklaşım, tablolara detay kimliğini belirten kolonların eklenmesidir. Örneğin kenarları ("edge") içeren tabloya kenarın ait olduğu karayolu ismini belirtmek üzere bir kolon eklenebilir. Ancak bu yaklaşım, aynı basit detay ya da geometrik elemanın birden çok karmaşık nesne tarafından paylaşılmasını gerektiren durumlarda yetersiz kalır; Karayolu ağlarının karmaşık nesneler olarak tanımlanması ve aynı bir kenarın farklı karayolu ağlarına ait olması durumunda olduğu gibi. Bu açıdan daha uygun olan yaklaşım, "ilişkisel birleştirme" yoluyla karmaşık nesnelerin oluşturulmasıdır [16].

3.3.1.7 Nesne Paylaşımı

DIGEST, hem detay hem de topolojik (konumsal) varlıklar düzeyinde nesne paylaşımına olanak tanımaktadır. Topolojik varlıklar düzeyinde, spaghetti veri yapısında nesne paylaşımı uygulanamaz. Çünkü bu veri yapısında, detaylarla topolojik varlıklar arasında yalnızca bire-bir ilişkiye izin verilmektedir. Ancak chain-node modunda her bir detay bir ya da daha çok geometrik varlığa işaret edebilir. Bir geometrik varlık ta, bir ya da daha çok detay tarafından işaret edilebilir. Diğer bir ifadeyle, detaylar ve geometrik varlıklar arasında çoktan-çoğa ilişkiler gösterim edilebilir. Detaylar düzeyinde, 3.3.1.6 da belirtildiği gibi karmaşık nesneler, DIGEST-A ve DIGEST-C'de detaylar bazında tanımlanabilmektedir. Dolayısıyla, karmaşık bir detay DIGEST-A da kayıt numaraları, DIGEST-C'de de anahtar değerler üzerinden kendisini oluşturan diğer detaylara işaret edebilir.

3.3.1.8 Meta Veri

DIGEST veri kalitesi için bir dizi tanımlayıcı sunmaktadır. Bunlar ile verinin kaynağı, doğruluğu, güncelliği, tutarlılığı, ve tamlığı belirtilebilmektedir. DIGEST ayrıca, transfer veri grubunun konumsal kapsamını aşan detayların bu durumlarının, verinin gizlilik derecesinin, ve başka kaynaklara açıklık derecesinin belirtilebilmesi için de tanımlayıcılar

içermektedir. Söz konusu tanımlayıcılar numerik değer ya da metin olarak belirtilebilmektedir.

DIGEST'te meta veri, transfer veri grubunun tamamı için genel olarak ya da tek tek detay, öznitelik, ve topolojik varlıklar düzeyinde belirtilebilir [144].

3.3.1.9 Nesne Bazında Veri Değişimi

DIGEST ile nesne bazında veri değişimi olanaklıdır. Nitekim DCW verisi seçmeli olarak transfer edilebilir. Ancak DIGEST dosya bazında veri değişimine yönelik olarak tasarlanmıştır [15], [138]. Bu nedenle, DIGEST veri değişim yapısı gereği bir transfer veri grubunda transfer verisi hakkında bulunması gereken bilgi düzeyi, nesne bazında veri transferini oldukça hantal kılacaktır. Söz konusu bilgi kapsamının eldeki transfer veri grubuna bağlı olarak değişebilmesi, yani bir kısım bilgilerin seçmeli olması bu sonucu değiştirmeyecektir. Çünkü bilgilerin seçmeli olarak transfer veri grubunda yer alabilmesi kodlama ve kod çözümleme işlemlerinin yükünü bir dereceye kadar hafifletse de, bu işlemler zorunluluk/seçmelilik durumunu dikkate almak zorundadır. O nedenle bu açıdan kabul edilebilir çözüm, veri modelinin başlı başına nesne bazında veri transferine yönelik bir destek sunmasıdır.

3.3.1.10 Zamansal Veri

DIGEST, zamansal veri desteği sunmamaktadır.

3.3.1.11 Multimedya

DIGEST, multimedya desteği sunmamaktadır.

3.3.1.12 Genişletilebilirlik

DIGEST karmaşık nesne kavramını desteklediği için herhangi bir nesne modellenemez. Ayrıca DIGEST'in, detay ve özniteliklerin kodlanması için kullandığı FACC (Feature and Attribute Coding Catalogue) katalogu genişletilebilmekte, yeni detay tanımları eklenebilmektedir. Ancak DIGEST, SAIF ve SDTS gibi genel amaçlı formatlara göre daha dar kapsamlı, özel bir format olduğu için, hem kullanılan veri tipleri de sınırlıdır hem de yeni veri tiplerinin kullanıcılar tarafından kolaylıkla eklenebilmesine yönelik bir desteği yoktur. Dolayısıyla bu tip genişletmeler ancak belirli bir otorite—DGIWG gibi—tarafından

yapılabilir. FACC açısından da durum aynıdır; kataloga ekleme ya da düzeltmeler, farklı kullanıcı gereksinimleri doğrultusunda DGIWG tarafından yapılabilir.

3.3.2 Gerçekleştirim

Bir DIGEST transer veri grubu("transmittal"), o transfer veri grubuna özel bir takım bilgiler içeren bir dosya (Transmittal Header File, THF) ile, bir ya da daha çok veri grubundan oluşur. Bir veri grubu (dataset) da, veriyi tanımlayan (Header Data Subset, HDS) ve veriyi içeren (Geo Data Subset, GDS) alt gruplarından oluşur (Şekil 38). THF, transfer ortamı, transfer veri grubunda yer alan veri grubu sayısı, verinin gizlilik derecesi ve güncelliği gibi bilgiler içerir. THF ayrıca, transfer veri grubunda yer alan veri gruplarının konumsal kapsamalarını ve hangi tip veri—vektör, raster, veya matris—içerdiklerini belirtir.

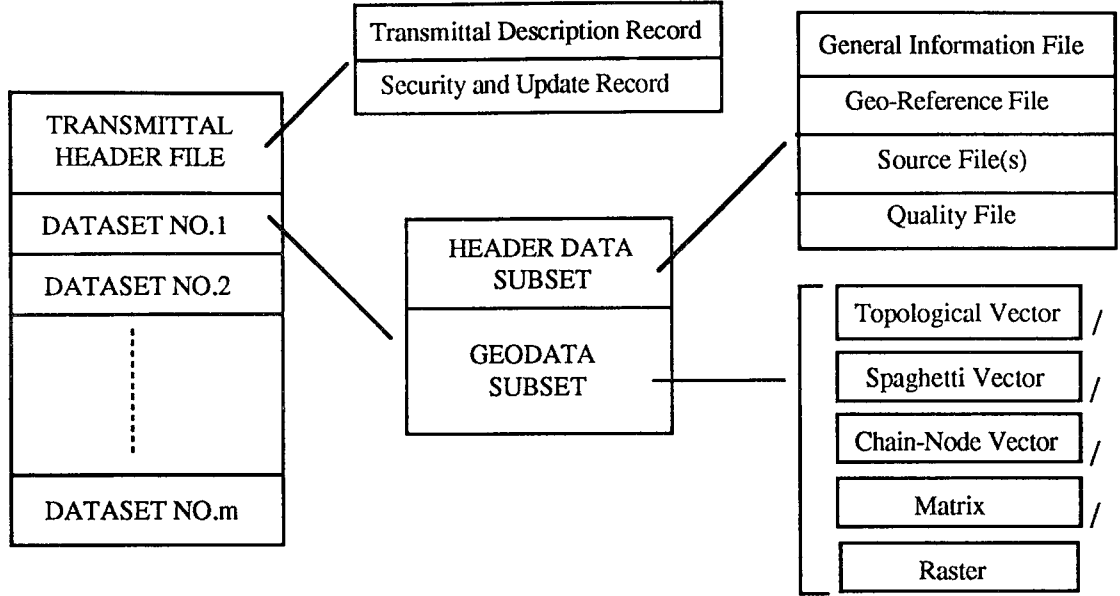
HDS, Şekil 38 de görüldüğü gibi dört tür dosya içerir. Bu dosyalar izleyen veri grubu için projeksiyon, veri kalitesi, veri tipi, referans sistemi, konumsal kapsam gibi bilgilerle, veri grubunda yer alan detay ve topolojik varlık sayılarını ve daha pekçok tanımlayıcı bilgiyi içerir.

GDS, transfer verisini içerir. Bu veri DIGEST'in desteklediği veri yapılarından birinde olmalıdır. Diğer bir ifadeyle, aynı bir veri grubu içerisinde farklı veri yapıları birlikte kullanılamaz. Bu durum Şekil 38 de "/" işareti ile belirtilmektedir. Diğer bir sınırlama, her bir veri grubu için yalnızca tek bir referans sistemi ve projeksiyonun geçerli olabilmesidir.

DIGEST ile hem nesne-tabanlı hem de ilişkisel (kayıt-tabanlı) bir gerçekleştirim uygulanabilir. Nesne-tabanlı gerçekleştirimde detaylar detay kayıtları ve diğer ilişkili kayıtlara göstergeçler yardımı ile gösterim edilir. Bu durumda GDS'te bulunabilecek kayıtlar, basit detay, karmaşık detay, topolojik varlık ve metin kayıtları ile detaylarla öznitelikleri ilişkilendiren kayıtlardır. İlişkisel gerçekleştirimde ise, transfer verisi ilişkisel tablolarda tutulur. Bu yaklaşımın amacı, alıcının doğrudan—kod çözümleme işlemine gerek olmadan—transfer verisi üzerinde işlem yapabilmesini sağlamaktır. DIGEST'te bu tarz gerçekleştirimin formatı "Vektör Relational Format" (VRF) olarak anılır. VRF hakkında daha ayrıntılı bilgi aşağıda verilmektedir.

3.3.2.1 Profiller

DIGEST'te, SAIF ve SDTS formatlarında tanımlandığı gibi bir profil kavramı yoktur. Bu, DIGEST'in, SAIF ve SDTS'e göre çok daha dar kapsamlı olmasının bir sonucudur.



Şekil 38. DIGEST veri değişim yapısı.

Bununla birlikte DIGEST, gerçekleştirimde bir dereceye kadar özellikli transferler için uyarlanabilirlik olanağı sunmaktadır. Yukarıda belirtildiği gibi, belirli bir transfer için nesne-tabanlı ya da kayıt-tabanlı bir gerçekleştirim uygulanabilmektedir. DIGEST ayrıca, belirli bir transferde en uygun veri yapısının belirtilebilmesi için bir dizi seçenek sunmaktadır. Bu seçenekler vektör veri için dört, raster ve matris veri için ikişer olmak üzere toplam sekiz adettir [144].

3.3.2.2 Detay Kodlama

DIGEST, detay ve özniteliklerin kodlanması için, bu amaçla yine DGIWG tarafından geliştirilmiş bir katalog (Feature and Attribute Coding Catalogue, FACC) kullanır. FACC ile amaçlanan, konumsal veri yönetiminde söz konusu olan detay ve öznitelikler için standart bir sınıflama ve kodlama sistemidir. FACC, DIGEST dökümanının 4. bölümünü oluşturmakla birlikte, yalnızca DIGEST'e yönelik olarak geliştirilmemiştir. O nedenle, FACC kendi başına ayrı bir standart olarak düşünülebilir.

FACC'ta her bir detay, 5 karakterlik tek anlamlı bir kod ile kimliklendirilir. İlk karakter, detay sınıfına karşılık gelir ve A dan Z ye alfabetik değerler alır. FACC'ta şimdilik 10 detay sınıfı bulunmaktadır. Bunlardan biri, belirli bir transfer veri grubuna özel detayları belirtmek için ayrılmıştır ve "S" harfi ile işaretize edilir. "S" sınıfı dışındaki diğer 9 sınıf için, bir alt sınıflandırma daha uygulanmıştır. Her bir alt sınıfta, yine A dan Z ye karakterler kullanılabilir. Alt sınıfı belirten karakter, 5 karakterlik kodun ikinci karakterini oluşturur. Kalan üç karakter 000 ile 999 arasında bir değer olabilir. Bir örnek olarak, FACC'ta Kamp

Alanı "AK060" kodu ile tanımlanır. Burada "A", "Kültür" ana sınıfını, "K" ise Kültür alt kategorilerinden biri olan "Rekreasyon" sınıfını simgelemektedir.

Doğal olarak, FACC'ın konumsal veri yönetiminde karşılaşılabilecek bütün detay ve öznitelik tanımlarını kapsamaması olanaksızdır. Bu nedenle DIGEST, belirli bir transfer için FACC'daki tanımlamaların yeterli olmaması durumunda, o transfere özel detay ve öznitelik tanımlamalarının bir veri sözlüğü ile transferine de olanak tanımaktadır. İdeal olarak, yeni detay ve öznitelik tanımlamalarının zaman içerisinde sürekli eklenmesi ile, FACC'ın bir noktada veri sözlüğüne gerek kalmadan bütün uygulamalara cevap verebilmesi hedeflenmiştir. Bu nedenle, FACC'ın belirli bir coğrafi ürün ya da veri tabanından bağımsız ve genişletilebilir olmasına ayrı bir özen gösterilmiş, ekleme ya da düzeltmelerin FACC'ın orjinal tasarımının da uyduğu bir takım kurallara uygun olarak yapılması öngörülmüştür [144]. Bugün bir çok ülke, ulusal kodlama standardı olarak FACC'ı uyarlama çabası içerisinde [124]. Örneğin Almanya'nın ATKIS (Authoritative Topographic Kartographic Information System) sisteminde FACC kullanılmıştır. Türkiye'de de Harita Genel Komutanlığı, 1:25 000 ve 1: 10 000 ölçekli haritalarda yer alan detay ve öznitelikler için FACC'a dayalı bir kodlama katalogu oluşturmuştur.

DGIWG, 1995'te FACC'ı gözden geçirerek çok daha kapsamlı bir katalog hazırlayacağını duyurdu [23]. Öte yandan, bütün uygulamalara cevap verecek çok geniş bir katalog, bir takım sakıncaları da beraberinde getirir [79]. Böyle bir katalogun oluşturulması, her şeyden önce uzunca bir süreç gerektirir. İkinci olarak, böyle bir katalogun organizasyonu—sürekli güncelleştirilmesi vs.—oldukça külfetli, kullanımı da bir o kadar güç olacaktır. Nihayet, çok geniş bir katalog işlemsel verimi oldukça olumsuz yönde etkileyecektir.

3.3.2.3 Kodlama Yöntemi

DIGEST, transfer verisinin kodlanması için üç seçenek sunmaktadır. Bunlardan ilk ikisi, ISO standartları olan ISO 8211 ve ISO 8824 (ASN.1) dir. Üçüncü seçenek, DIGEST'e özel olarak geliştirilmiş olan VRF (Vector Relational Format) dir. DCW'nin formatı olarak bilinen VPF (Vector Product Format) formatı ile VRF aynıdır [143]. ISO 8211 ve ISO 8824 bir önceki bölümde ve yukarıda detaylı olarak incelendiği için, burada yalnızca VRF üzerinde durulacaktır.

VRF, ilişkisel veri modeline ve temel geometrik elemanlara dayalı vektör veri gösterimine yönelik bir formattır. Bugün kullanımda olan CBS'lerin çoğunun ilişkisel veri modeline dayanması ve DIGEST veri modelinde olduğu gibi temel geometrik elemanları

kullanması—"Geo-relational model"—nedeniyle, VRF'in potansiyel kullanım alanı oldukça geniştir. VRF'in ardındaki tasarım felsefesi, alıcıya transfer edilecek veriyi bir ara formata ya da değişim formatına dönüştürmeden önce, tarayarak—tercihen bir GKA yardımıyla—inceleyebilme olanağı sunmak ve ancak ona göre varsa, gereksinim duyduğu veriyi transfer etmesini sağlamaktır.

VRF'te detay, öznitelik, ve geometrik elemanlar ilişkiler (ilişkisel tablolar) kullanılarak organize edilir. Dolayısıyla bir detaya ait veri diğer detay, öznitelik, ve geometrik eleman tablolarına dağılmış olabilir. Detay, öznitelik, ve geometrik eleman tabloları arasında ilişki, anahtar alanlar yardımıyla sağlanır. Bu yolla, detaylar ve geometrik elemanlar arasında bire-bir, bire-çok, ve çok-çok ilişkiler gösterim edilebilir. Dolayısıyla, bir detaya ait veri "ilişkisel birleştirme" yoluyla bir araya getirilebilir.

VRF, gerek konum ve gerekse diğer özniteliklere göre veriye doğrudan erişim olanağı sağlar. Bunun için VRF'te detaylar konumsal kapsam ve özniteliklerine göre indekslenir. Söz konusu konumsal indexleme, veri gruplarının daha küçük boyutlu alt gruplar halinde organizasyonuna olanak tanımaktadır. Diğer bir ifadeyle, belirli bir yatay bölümlene sistemine göre oluşturulan bölümler arasında geçiş sağlanabilir [144].

3.3.3 Değişim Ortamı

DIGEST-C, üreticiden kullanıcıya veri transferine yönelik olarak geliştirilmiş, bu formatta dağıtımı yapılacak verinin etkin bir biçimde kullanıcıya sunulması amaçlanmıştır. O nedenle DIGEST-C formatında kullanıma sunulan DCW ürünü, kullanıcının DCW verisini grafik görüntü ekranında görüntüleme, sorgulama, ve transfer etmesini sağlayacak bir GKA ile desteklenmiştir. Bu DCW ve dolayısıyla DIGEST'e yönelimi önemli ölçüde teşvik etmiştir.

DIGEST-A için geliştirilmiş bir UPA henüz bulunmamakla birlikte, DGIWG içerisinde Kanada tarafından başlatılan bir "Digest View" projesi vardır [142]. Bu proje ile, DIGEST formatındaki raster ve vektör veriyi görme, sorgulama, ve transfer etmeyi sağlayacak bir yazılım hedeflenmiştir. Bu yazılım farklı platformlarda (DOS, UNIX, MAC) çalışabilir ve kullanıcılara (kamuya) açık olacaktır. Kuşkusuz, böyle bir yazılım DIGEST formatında gelen ürünlerin kullanımını önemli ölçüde teşvik edecek ve DIGEST'in yaygınlaşmasını sağlayacaktır.

3.3.3.1 Dokümantasyon

DIGEST dokümantasyonu dört bölüm olarak düzenlenmiştir. Birinci bölüm, formatın genel bir tanımını içerir. İkinci bölümde, DIGEST veri modeli, veri değişim yapısı, ve fiziksel kodlama ele alınmaktadır. Üçüncü bölümde kodlar, parametreler, ve etiketler yer alır. Detay ve öznitelik kodlama katalogu (FACC) ise, dördüncü bölümü oluşturmaktadır.

DIGEST dökümanı kolaylıkla anlaşılabilir değildir. Bunda, DIGEST'in çok uluslu bir çalışma grubu tarafından hazırlanmış olmasının rolü büyüktür [124]. Bunun sonucu olarak terminolojide tutarsızlıklar, dokümantasyonda gerek kapsam ve gerekse organizasyon açısından çeşitli eksiklikler söz konusudur. Örnek olarak, DIGEST dökümanın en başında etraflı bir giriş yapılmamış olması, ISO 8211 ve ISO 8824 ile ilgili kısımlara bu standartlara ilişkin genel bilgilerle başlanmamış olması, VRF gibi yeni bir formatın gerekçesinin yeterince açıklanmamış olması, farklı topolojik düzeyler arasındaki ayırımın yeteri açıklıkta belirtilmemiş olması, projeksiyon sistemleri ve datumlar hakkındaki bilgilerin teorik model kapsamı dışında tutulmuş olması verilebilir.

3.3.3.2 Servisler

DIGEST, servisleri kapsamamaktadır. Gerek DCW kapsamındaki GKA ve gerekse Digest View ile amaçlanan, DIGEST formatındaki verinin alıcı tarafından sorgulanması ve transferi için gerekli GKA desteğini sunmaktır. Ancak tersi yönde, herhangi bir formattaki verinin DIGEST formatına dönüştürülmesi için de, çeviri rutinleri ve araçlara gereksinim vardır. DIGEST için böyle bir destek henüz mevcut değildir.

3.3.3.3 Değişim Ortamı Araçları

DIGEST-C formatındaki verinin seçmeli transferi olanaklıdır. Çünkü VRF'in ilişkisel veri modeline dayanması ve indeksler kullanması seçmeli transfer için gerekli alt yapıyı sağlamaktadır. Geriye VRF verisini görme, sorgulama, ve transfer etmeyi sağlayacak destekleyici yazılımlar kalmaktadır. DCW ürünü bu desteği sunmakta ve Dünya genelindeki temel konumsal ve konumsal olmayan veri yanında, PC ortamı için geliştirilmiş bir GKA da içermektedir. Bu GKA ile DCW verisi, grafik görüntü ekranında görüntülenerek, sorgulanabilmekte, ve nihayet transfer edilebilmektedir. Değişim formatlarının ve konumsal veri ürünlerinin destekleyici yazılımlarla birlikte kullanıma sunulması bakımından, DCW önemli bir aşama olmuştur.

Yukarıda da belirtildiği gibi, DIGEST-A için henüz böyle bir GKA bulunmamaktadır. Digest View projesinin tamamlanmasıyla, DIGEST-A formatındaki verinin görüntülenmesi, sorgulanması, işlenmesi ve transfer edilmesi sağlanabilecektir. DIGEST çevricilerinin derleyici/yorumlayıcı teknolojisi ile geliştirilmesi yönünde Yazar' ın bilgisi dahilinde olan bir çalışma bulunmamaktadır.



4. İRDELEME VE DEĞERLENDİRME

Bu bölümde 6. bölümde elde edilen bulgular irdelenmektedir. Bu bölüm iki kısım olarak düzenlenmiştir. Birinci kısımda, irdemeler özetlenmekte ve Türkiye için UKVA gereksinimlerini karşılayabilecek bir konumsal veri değişim standardı önerilmektedir. İkinci kısımda ise, seçilen standarda Türkiye'ye özel olarak yapılması gerekli genişletmelere değinilmektedir.

4.1 İrdeleme ve Değerlendirme

Bu kısımda, 6. Bölümde elde edilen bulgular irdelenmekte ve değerlendirmeye alınan üç formatın, her bir ölçüt açısından değerlendirilmesi yapılmaktadır. Söz konusu değerlendirme ayrıca, Tablo 2 de özet olarak verilmiştir.

4.1.1 Veri Modeli

Veri modeli tipi: SAIF, diğer iki formatın dayandığı nesne-tabanlı ve ilişkisel modellere göre daha yüksek düzeyli bir model olan, Nesneye-Dayalı veri modeline dayanmaktadır. Dolayısıyla SAIF, DIGEST ve SDTS'e göre çok daha genişletilebilir bir veri modeline sahiptir. Çeviri, kodlama ve kod çözümleme rutinleri SAIF veri modeline eklenebilir.

Raster veri: Raster veri açısından, SAIF oldukça kapsamlı bir desteğe sahiptir. DIGEST te, bazı eksikleri olmakla birlikte (örn. dörtlü ağaç raster veri temsili) çeşitli raster veri temsillerini desteklemektedir. SDTS'te ise, şimdilik yalnızca en çok kullanılan dört raster temsili tanımlanmıştır. Başka raster veri temsilleri için yapılması gerekli eklemeler SDTS'te en fazladır.

Kartoğrafik temsil: Bu açıdan en güçlü SDTS, en zayıf ta DIGEST dir. SAIF yeterli destek sunmakta ve sembol kitaplıklarının belirtilebilmesine olanak tanımaktadır. Her üç format da, en çok kullanılan projeksiyon sistemlerini desteklemekte ve kullanıcıya özel projeksiyonların tanımlanmasına olanak tanımaktadır.

Koordinatlar: SDTS ve DIGEST yalnızca 2 ve 3 boyutlu koordinat belirtmeye izin verirken, SAIF zaman boyutunun da belirtilmesine olanak tanımaktadır. Koordinat çözünürlüğü açısından, SDTS ve DIGEST'te çözünürlük pratik olarak sınırsızdır. SAIF tasarımı, tamsayı değerler için 64, kesirli sayı değerler için ise 80 bit lik çözünürlük öngörmüştür.

Topoloji-Geometri Ayırımı: Her üç format ta, topoloji-geometri ayırımını desteklemektedir. Ancak, SAIF veri modeli bu bakımdan çok daha düzenlidir. Ayrıca SAIF'te topoloji daha yüksek düzeyde—nesne bazında ilişkiler—ve daha kapsamlı olarak tanımlanabilmektedir.

Karmaşık Nesneler: SAIF ve DIGEST, detay düzeyinde karmaşık nesne tanımlamaya olanak tanır. Buna göre karmaşık detaylar, daha basit detaylardan ya da yine karmaşık detaylardan oluşur. SDTS'te ise, karmaşık detaylar geometri düzeyinde tanımlanır. SDTS'de böyle bir tanımlama "karmaşık nesne" olarak anılır. Buna göre, bir karmaşık detayın geometrisi, ilgili karmaşık nesne tarafından gösterim edilir. Bu durumda, bir karmaşık detayı oluşturan diğer detaylara ait konumsal olmayan öznitelik bilgisinin elde edilmesi, SAIF ve DIGEST-A'da olduğunun aksine, ayrı bir arama gerektirir. Bu tür bir arama, karmaşık nesnelerin ilişkisel birleştirme yoluyla tanımlandığı DIGEST-C'de de gereklidir.

Nesne Paylaşımı: Nesne paylaşımını DIGEST ve SAIF detay, SDTS ise geometri düzeyinde desteklemektedir. Ancak, nesne paylaşımının SAIF ve DIGEST'te daha yüksek bir düzeyde tanımlanmasının, transfer veri grubunun boyutu açısından bir önemi yoktur. O nedenle bu ölçüt bazında, formatlardan birinin daha üstün olduğunu söylemek çok güçtür.

Meta Veri: Her üç format ta meta veri desteğine sahiptir. Her üç format ta meta verinin genel olarak, transfer veri grubunun tamamı için ya da tek tek detaylar bazında belirtilmesine olanak tanımaktadır.

Nesne Bazında Veri Değişimi: SAIF nesne bazında veri değişimini desteklemektedir. DIGEST ve SDTS ise, bütün bir dosyanın transferine yönelik olarak tasarlanlanmış olmaları nedeniyle, nesne bazında veri değişimi için uygun değildir.

Zamansal Veri: SDTS ve DIGEST veri modellerinin zamansal veri gösterimine yönelik bir desteği yoktur. Buna karşılık SAIF, zamansal veri için oldukça güçlü bir desteğe sahiptir.

Multimedya: Her üç format da multimedya desteğinden yoksundur.

Geniřletilebilirlik: SAIF, DIGEST ve SDTS'ye gre ok daha yksek bir geniřletilebilirlik sunmaktadır. Bunun nedeni SAIF'in, ıkıř noktası "geniřletilebilir yazılımlar retmek" olan Nesneye-Ynelim teknięini benimsemesidir. Bu nedenle, SAIF Nesneye-Dayalı Veri Modeli'ni kullanmaktadır. SAIF Standart řeması, konumsal veri gsterimi iin adeta bir "alt yapı" sunmaktadır. Bu alt yapı, zel gereksinimlere gre kolaylıkla zenginleřtirilebilir ve yeni veri tipi tanımlamaları standart řemaya eklenebilir. Buna karřılık SDTS ve DIGEST, alıřılagelmiř veri modelleri olarak bilinen Varlık-İliřki ve İliřkisel veri modellerini ve yine alıřılagelmiř gerekleřtirim tekniklerini kullanmaktadırlar. O nedenle, geniřletmelerin yerine getirilmesi, gerek veri modeli ve gerekse geniřletmelerin mevcut eviriciler zerindeki etkisi bakımından, SAIF'tekine oranla ok daha g olacaktır. Bu durumda da geniřletme, hi te pratik olmayan bir m yolu olabilir.

4.1.2 Gerekleřtirim

Profiller: SAIF ve SDTS ok genel standartlar oldukları iin profiller yoluyla gerekleřtirmeyi desteklemektedir. SAIF iin 4, SDTS iin ise 2 profil ngrlmřtr. Bu bakımdan, SAIF'in desteęi daha gldr. Bu profillerden yalnızca ilk SAIF profili hazır ve kullanımdadır. SDTS profilleri henz tamamlanamamıřtır. Ayrıca, SAIF iin DIGEST ve SDTS iin DIGEST ve DX-90 profilleri geliřtirilmektedir. DIGEST, SAIF ve SDTS'e gre ok daha dar kapsamlı bir standart olduęu iin profil kavramı bulunmamaktadır.

Detay kodlama: DIGEST detay kodlama iin FACC'ı kullanmakta ve zellikli detay tanımları iin veri szlę kullanımına izin vermektedir. SDTS'in kendi i katalogu bulunmakla birlikte, dıř katalog kullanımına da izin vermektedir. SDTS'nin oęu gerekleřtirimlerinde kendi katalogunun kullanılmadıęı belirtilmektedir [137]. SDTS'te de zellikli veri tanımlamaları veri szlę yardımıyla belirtilebilmektedir. SAIF ND bir gerekleřtirmeyi benimsedięi iin, alıřılagelmiř detay kodlama yaklařımını desteklememektedir. SAIF gerekleřtiriminde de bir tr kodlama gereklidir ama bunun iin kodlama kataloglarına gerek yoktur. Transfer veri grubu, resmi bir dilde zellikli detay tanımlamalarını da ierdięi iin, veri szlęne gerek yoktur. Bununla birlikte SAIF, dıř kodlama kataloglarının kullanımına da izin vermektedir.

Kodlama Yntemi: DIGEST'te veri, ISO 8211, ISO 8824, ve VRF formatlarında kodlanabilmektedir. Dolayısıyla DIGEST hem evrim-ii hem de evrim-dıřı veri transferini desteklemektedir. VRF te sayısal deęerler iin ikili ve karakterler iin ASCII kodlama birarada uygulanabilmektedir. SAIF'te kodlama formatı SAIF/ZIP tir. Etiketsiz, ASCII kodlama ve sıkıřtırma uygulanır ve evrim-ii transfer desteklenmektedir. SDTS, ISO 8211'i kullanır. ASCII ya da ikili, etiketli kodlama uygulanır. ISO 8211, veri ile birlikte veri

tanımını da içerdiğinden alıcının sunucu verisinin soyut sözdizimini bilmesi gerekmez. Ancak, ISO 8211 çevrim-içi transfer için uygun değildir. Dolayısıyla kodlama yöntemi açısından SAIF ve DIGEST daha avantajlı görünmektedir.

4.1.3 Değişim Ortamı

Dokümantasyon: SAIF dokümantasyonu oldukça iyidir. Kavramlar iyi açıklanmış, veri modeli ve gerçekleştirim arasındaki ayırım özellikle vurgulanmıştır. Kullanılan terminoloji standart terminolojidir. SDTS dokümantasyonu kolayca anlaşılabilir değildir. Bunun nedenlerinden biri, veri modeli ve gerçekleştirim kavramları arasındaki ayırımın yeterince belirgin olmamasıdır. Diğer neden, "standart" olarak önerilen bir formatın, standart olmayan bir terminoloji ile sunulmasıdır. DIGEST dokümanının organizasyonu kötüdür. Çok uluslu bir çalışma grubu tarafından hazırlanmış olması nedeniyle tutarsızlıklar içermektedir. Veri modeli ve gerçekleştirim ayırımı yeterince açık değildir.

Servisler: Her üç format için de servis desteği, farklı açılardan yetersizdir. SAIF veri modelinde servisler bulunmamakla beraber, istendiğinde çevirici kodlayıcı ve kod çözücü rutinler eklenebilir. Bununla birlikte, SAIF çeviricilerinin geliştirilmesini kolaylaştırmak amacıyla geliştirilmiş olan SAIF Toolkit, SAIF formatında veri okunabilmesi ve yazdırılabilmesi için gerekli rutinler içermektedir. SDTS için, SDTS verisinin ISO 8211'de kodlanması/kod çözümlenmesi için gerekli rutinleri içeren "FIPS 123 rutin kitaplığı" bulunmaktadır. SDTS çevirici geliştirilicilerine yönelik daha geniş kapsamlı bir destek için geliştirilmiş olan "CSP" (Common Software Platform) "toolkit"i, henüz prototip aşamasındadır. DIGEST, servisleri kapsamamaktadır. Gerek DCW kapsamındaki GKA ve gerekse "Digest View" ile amaçlanan, DIGEST formatındaki verinin alıcı tarafından sorgulanması ve transferi için gerekli GKA desteğini sunmaktır. Ancak tersi yönde, herhangi bir formattaki verinin DIGEST formatına dönüştürülmesi için gerekli rutinler bulunmamaktadır.

Değişim Ortamı Araçları: SAIF Toolkit, derleyici/yorumlayıcı teknolojisi ile SAIF çeviricilerinin geliştirilebilmesi için gerekli desteği sunmaktadır. SAIF'te, veri tanımlama dili olarak "SAIFtalk" bulunmaktadır. Seçmeli transfer desteği henüz yeterli değildir. Çünkü SAIF Toolkit uygulama programcılarına yöneliktir. Son kullanıcılara gönderici verisi üzerinde sorgulama ve seçim olanağı sağlayacak bir GKA ya yönelik "SAIFTools" projesinin, 1996 başlarında tamamlanması beklenmektedir. DIGEST çeviricilerinin derleyici/yorumlayıcı teknolojisi ile geliştirilmesi yönünde Yazar' ın bilgisi dahilinde olan bir çalışma bulunmamaktadır. DIGEST-C'de VRF'in ilişkisel veri modeline dayanması ve indeksler kullanması seçmeli transfer için gerekli alt yapıyı sağlamaktadır.

Tablo 2. DIGEST, SAIF, ve SDTS'in değerlendirme ölçütlerine göre incelenmesinden elde edilen sonuçlar.

<i>Format</i> <i>Ölçüt</i>	DIGEST	SAIF	SDTS
Veri modeli tipi	Varlık-Tabanlı (DIGEST-A) ve İlişkisel (DIGEST-C)	Nesneye-Dayalı veri modeli, çoklu miras devralma destekleniyor	Varlık-Tabanlı ("entity-attribute-attribute value") Nesneye-Dayalı değil
Raster veri	Çeşitli raster gösterimleri desteklenmekte, bazı eksikler var (örn. dörtlü-ağaç gösterimi)	Oldukça geniş bir raster veri desteği var	Şimdilik yalnızca dört çeşit raster veri gösterimi tanımlanmış
Kartografik gösterim	Mevcut destek oldukça zayıf	Yeterince güçlü bir destek işaret kitaplıkları kullanılabilir	Oldukça güçlü bir destek
Koordinatlar	Koordinatlar yalnızca 2 ve 3 boyutlu olarak belirtilebilmekte, 4. boyut desteklenmemekte, pratik olarak sınırsız çözünürlük	Koordinatlar 1, 2, 3, ve 4 boyutlu olarak belirtilebilir, tasarım, 64-80 bit lik çözünürlük öngörmekte	Koordinatlar yalnızca 2 ve 3 boyutlu olarak belirtilebilmekte, 4. boyut desteklenmemekte, pratik olarak sınırsız çözünürlük
Topoloji-geometri ayırımı	Destekleniyor	Destekleniyor	Destekleniyor
Karmaşık nesneler	Destekleniyor	Destekleniyor	Destekleniyor
Nesne paylaşımı	Destekleniyor (detay bazında)	Destekleniyor (detay bazında)	Destekleniyor (geometri bazında)
Meta veri	Destekleniyor; Meta veri genel ve özel olarak belirtilebilir	Destekleniyor; Meta veri genel ve özel olarak belirtilebilir	Destekleniyor; Meta veri genel ve özel olarak belirtilebilir
Nesne bazında değişim	Desteklenmiyor	Destekleniyor	Desteklenmiyor
Zamansal veri	Desteklenmiyor	Destekleniyor	Desteklenmiyor
Multimedya	Desteklenmiyor	Desteklenmiyor	Desteklenmiyor
Genişletilebilirlik	Sınırlı	Yüksek derecede genişletilebilir	Sınırlı
Profiller	Profil kavramı yok, çünkü DIGEST "özel" (tanımlanmış) bir standart	Destekleniyor, vektör ve raster veri için 4 profil öngörülmüş, bunlardan birincisi hazır, diğerleri ve bir de DIGEST profili geliştirilmekte	Destekleniyor, vektör (TVP) ve raster profilleri öngörülmüş, Her iki profil ve ayrıca DIGEST ve DX-90 profilleri geliştirilmiş
Detay/Öznitelik kodlama	FACC (Feature and Attribute Coding Catalogue) kullanılmakta	Alışlagelmiş detay kodlama yaklaşımı desteklenmemekte, ancak FACC gibi "dış" kodlama katalogları da kullanılabilir	SDTS'in kendi ("iç") kodlama katalogu yanında, "dış" kodlama katalogları da kullanılabilir
Kodlama yöntemi	ISO 8211, ISO 8824, ve VRF, ASCII ve ikili, etiketli kodlama, çevrim içi transfer destekleniyor	SAIF/ZIP; etiketli, ASCII kodlama, çevrim içi transfer destekleniyor	ISO 8211; etiketli, ASCII veya ikili kodlama, çevrim içi transfer uygulanabilir

Tablo 2. (devam)

Dokümantasyon	Anlaşılması zor, terminoloji kullanımı tutarsız, veri modeli ve gerçekleştirim kavramları karıştırılmış, genel düzenleme kötü	Özellikle SAIF 3.2 dokümantasyonu oldukça iyi	Anlaşılması zor, terminoloji kullanımında tutarsızlıklar var, veri modeli ve gerçekleştirim kavramları karıştırılmış
Servisler	Veri modeli bazında desteklenmiyor, Gerçekleştirim desteği henüz yok	Destekleniyor, şimdilik <i>SAIF Toolkit</i> ile, Mevcut SAIF Standart Şemasında servisler yok, ancak eklenebilir	Veri modeli bazında desteklenmiyor, Gerçekleştirim bazında CSP geliştiriliyor
Değişim ortamı araçları	Henüz desteklenmiyor; <i>DigestView</i> geliştirilmekte, DCW GKA destekli	Destekleniyor; <i>SAIF Toolkit</i> ve <i>SAIF Tools</i> , hazır, ayrıca <i>SAIFTalk</i> Veri Tanımlama Dili var	Henüz desteklenmiyor

Ancak VRF verisini görme, sorgulama, ve transfer etmeyi sağlayacak destekleyici yazılımlar eksiktir. Sürmekte olan "Digest View" projesi ile, bu eksikliğin giderilmesi amaçlanmıştır. SDTS'nin değişim ortamı araçları desteği çok zayıftır. Derleyici araçları bulunmamaktadır. VTD olarak BNF kullanılabilir. Seçmeli transfer araçları bulunmamaktadır. Sonuç olarak, değişim ortamı araçları açısından en güçlü desteği sunan SAIF tir.

4.2 İrdelemeler

Veri modeli açısından en uygun format SAIF tir. Tablo 2 den görülebileceği gibi SAIF, veri modeli ölçütleri açısından SDTS ve DIGEST'ten daha üstündür. SAIF, ND bir veri modeline sahip olması nedeniyle, diğer formatlardan çok daha yüksek derecede bir genişletilebilirlik sunmaktadır. Bu, UKVA açısından çok önemlidir. Çünkü UKVA'da çok farklı kullanıcı grupları olacak ve zamanla yenileri eklenebilecektir. SAIF veri modelinin ND olmasının UKVA açısından diğer önemi, veri değişim servislerinin istendiğinde veri modeline eklenebilmesidir. DIGEST ve SDTS veri modelleri aynı şey için uygun değildir. Bu yolla, SAIF veri modeli genişletilebilir ve doğrudan çeviriye dayalı, ND bir veri değişim ortamının süper modeli olarak kullanılabilir. SAIF, ND bir veri değişim ortamı için diğer formatlardan çok daha uygundur.

Gerçekleştirim açısından profiller bazında formatlardan birinin kesin üstünlüğünden söz etmek güçtür. Detay kodlama bazında, FACC'ın SDTS'nin kendi kodlama katalogundan çok daha yaygın kullanım bulunduğu bir gerçektir. Bu açıdan SAIF yeni bir yaklaşım sunmakta ve alışlagelmiş detay kodlamayı desteklememektedir. Farklı kullanıcılara, detaylarını istedikleri gibi tanımlama olanağı sunmaktadır. Bu durumda çeviricinin hangi detayın hangi detaya, nasıl karşılık geldiğini bilmesi gerekir. Gerçi burada da çevirici öznitelik değerleri için bir kodlama uygulamalıdır ancak bu kodların belirlenmesi için gerekli

çaba, detay kodlama kataloglarının oluşturulmasındakinden çok daha azdır. Çünkü bu kodlar çevirici geliştirici tarafından belirtilir, katalog oluşturmadaki gibi tarafların bir araya gelerek detay tanımlamaları üzerinde anlaşması gerekmez. Bu önemli bir esnekliktir. Bu yaklaşım, alışlagelmiş kodlama kataloglarının geliştirilmesi ve yaşatılmasına göre daha kolaydır. Bu bakımdan SAIF üstün görünmektedir. Bununla birlikte, UKVA'da yer alacak tarafların kullandıkları detay/öznitelikleri tanımlamaları, yani kendi şemalarını oluşturmaları gerekir. Bu konuya aşağıda daha ayrıntılı olarak değinilmektedir. Kodlama yöntemi bazında, en çok seçenek sunan DIGEST dir. SDTS bu açıdan en zayıftır. SAIF kodlama yöntemi, çok yaygın olarak kullanılan bir sıkıştırma tekniğini kullanmakta ve özlü bir kodlamaya olanak tanımaktadır.

Değişim ortamı açısından üstün olan format yine SAIF tir; SAIF dokümantasyonu DIGEST ve SDTS den daha iyidir. Servisler bazında SAIF Toolkit ve SAIF Tools desteği vardır. Değişim ortamı araçları olarak SAIFTalk veri tanımlama dili ve SAIF Toolkit vardır. SDTS için geliştirilmekte olan CSP henüz prototip aşamasındadır. Bu yalnızca çevirici geliştirmeye yöneliktir. Değişim ortamı araçları bulunmamaktadır. DIGEST için de geliştirilmekte olan DigestView, DIGEST çeviricilerinin geliştirilmesine yönelik değildir, DIGEST formatındaki verinin gösterilmesi, taranması, sorgulanması ve ardından istenen verinin transfer edilmesine yöneliktir.

Sonuç olarak, UKVA için en uygun format SAIF tir. Böylece UKVA tarafları veri transfer (ithal/ihrac) işlemlerini SAIF formatı üzerinden gerçekleştireceklerdir. Tarafların tek bir formatta veri üretmeleri, UKVA'da konumsal veri değişimini çok önemli ölçüde kolaylaştıracaktır. Öte yandan, Konumsal veri değişim standartları alanında hala birlik sağlanabilmiş değildir. A.B.D.'de SDTS'nin standartlaşması için gösterilen çabaya ve Şubat 1994'den itibaren federal kuruluşlara getirilen SDTS formatında veri sağlama yasal zorunluluğuna rağmen, formata yönelimin çok hızlı bir biçimde olduğu söylenemez. DIGEST bir NATO standardı olması, geleneksel konumsal veri modellerine dayalı özellikli transferler için daha genel standartlar olan SDTS ve SAIF'e göre daha uygun olması nedeniyle bugün olduğu gibi yakın bir gelecekte de önemli bir rol oynayacaktır. DIGEST, ISO TC 211'in çalışmalarını da etkilemektedir. Diğer taraftan, standartlar alanında giriş bölümünde söz edilen uluslararası çabalar vardır. "Bu koşullar altında UKVA için SAIF'in önerilmesi ne kadar anlamlıdır" diye bir soru akla gelebilir.

Bu tezde değerlendirmeye alınan ve uluslararası arenada söz sahibi standartlar açısından bakıldığında, UKVA'da SAIF'in kullanılması, SDTS ve DIGEST'i geçersiz formatlar yapmayacaktır. SAIF, DIGEST ile doğrudan DIGEST profili üzerinden, SDTS ile de dolaylı olarak SDTS'nin DIGEST profili üzerinden uyumludur (bkz. bölüm 1).

Dolayısıyla DIGEST ve SDTS formatlarında gelen veri transfer edilebilir ve bu formatlarda veri üretilebilir. Böylece, UKVA'nın uluslararası uyumluluğu da sağlanmış olmaktadır.

Uluslararası organizasyonların standartlar alanındaki çalışmalarına gelince, bu çalışmaların kapsamı, bu tezde değerlendirilen formatlardan daha geniştir. Bu çalışmalar, veri tanımlamaları ve gösterimleri, yazılım ve donanımları farklı sistemlerin veri paylaşımını, interoperabilite açısından ele almaktadır (bkz. bölüm 1). Ancak, bu çalışmaların nasıl sonuçlanacağı ve elde edilen sonuçların ne kadar kabul göreceği belli değildir; ISO TC 211 oldukça yeni bir organizasyondur ve çalışmalarını sonuçlandırması uzunca bir süreç alabilir. CEN TC 287 ve OGC'nin çalışmalarında çeşitli gelişmeler yaşanmakla birlikte, tamamlanma süreleri belli değildir. Kaldı ki, bu çalışmaların üçünün de ayrı ayrı başarılı olması, üç yeni standart anlamına gelir. Bu durumda hangisi kullanılacaktır? En iyi olasılıkla, bu çalışmalardan tek bir interoperabilite standardının ortaya çıkması ve UKVA'da bu standarda geçilmesi durumunda önemli olan, bunun bu tezde elde edilen sonucu kullanacak UKVA'da yapacağı etkinin minimum olmasıdır. Bu açıdan da en uygun olan format yine SAIF'tir. Çünkü interoperabilite açısından DIGEST ve SDTS'ten daha uygun olan SAIF tir. Başarılı senaryo sonunda elde edilecek interoperabilite standardının SAIF yerine, her üç standardı da kullanması durumunda ise, söz konusu geçişin yapacağı etkinin, UKVA'da SDTS ve DIGEST kullanılması durumundakinden daha olumsuz olacağı da söylenemez. Sonuç olarak UKVA için SAIF'in önerilmesi, standartlar alanındaki uluslararası gelişmeler açısından da uygundur.

4.3 SAIF'in Uyarlanması

SAIF'in uyarlanabilmesi için, formatın Türkiye'ye özel gereksinimler için genişletilmesi gereklidir. Ayrıca UKVA için bir detay/öznitelik katalogu ve işaret kitaplıkları geliştirilmelidir.

SAIF'e yapılması gerekli başlıca genişletme koordinat sistemleri içindir. SAIF'te tanımlı yatay ve düşey referans sistemlerine, Türkiye'de kullanılan sistemleri eklemek gerekir. SAIF'te 19 projeksiyon türü tanımlıdır. Bunlar çok yaygın olarak ve Türkiye'de kullanılan UTM, Transverse Mercator, Lambert Konformal Konik gibi projeksiyonlardır. Bu projeksiyon türlerinin yeterli olmaması durumunda yeni bir projeksiyon türü SAIF'te kolaylıkla tanımlanabilir. Dolayısıyla SAIF, Türkiye'ye özel koordinat sistemlerinin SAIF standart şemasına eklenmesi için gerekli desteği sunmaktadır. Yapılması gereken, SAIF standart şemasında desteklenen ve Türkiye'de kullanılan sistemlerin karşılaştırılarak, varsa gerekli parametrelerin eklenmesidir.

Yukarıda belirtildiği gibi UKVA'da yer alacak tarafların kullandıkları detay/öznitelikleri tanımlamaları, yani kendi kataloglarını oluşturmaları gerekir. Burada söz konusu olan yalnızca farklı kurumların kendi detay tanımlamalarını belirlemeleridir. SAIF kullanılması nedeniyle, her kurumun üzerinde anlaşacağı ortak detay tanımlamalarına ve detay kodlarına gerek yoktur. SAIF tasarım ilkesi, farklı kurumların kullandıkları detayları gereksinimlerine en uygun bir biçimde tanımlama esnekliğini sunmaktadır. Daha önce de işaret edildiği gibi, örneğin TCK'nın karayolu tanımı ile HGK'nınki birbirinden oldukça farklı olacaktır. HGK'nın tanımında örneğin "yolun-son-durumu" özneliği bulunmayacaktır. UKVA kurumlarının detay/öznitelik tanımlamaları UKVA'da "bilgi sunucuları" için de gerekli olacaktır. Söz konusu tanımlamalar için, FACC'ın detay sınıflaması yaklaşımından yararlanılabilir. FACC uyumlu detay/öznitelik tanımlamaları UKVA'da DIGEST formatında veri transferi açısından kolaylık sağlayacaktır. Ayrıca, HGK'nın ulusal bir coğrafi Veri Tabanı (CVT) oluşturulmasına yönelik olarak 1: 25 000 ve 1: 10 000 ölçekli haritalarda yer alan detayları FACC'a göre tanımlama çalışmaları ile, DİE'nin standart bir detay kodlama katalogu oluşturma yönündeki çalışmaları yararlı olacaktır. UKVA kapsamında, söz konusu çalışmalar yeniden gözden geçirilmelidir.

UKVA'da konumsal veri değişimi için ayrıca, mevcut kartografik gösterim katalogunun gözden geçirilerek, yeni bir katalogun oluşturulması gerekir. İşaretlendirmede farklı bakış açıları olabilir ve farklı kullanıcı grupları aynı bir detayı farklı işaretlerle göstermek isteyebilirler. Ancak, UKVA'da standart bir kartografik gösterimin kullanılması çok önemli bir kolaylık sağlayacaktır. Bunun için, bütün kullanıcı gruplarının üzerinde anlaşacakları ortak bir kartografik gösterim katalogunun oluşturulması gerekir. Türkiye'de bugün kullanılmakta olan kartografik gösterim katalogu yetersiz kalmaktadır. Bunun bir nedeni, mevcut kartografik gösterim katalogunun yeterince ayrıntılı olmamasıdır. Bunun nedeni de, Türkiye'nin ayrıntılı bir detay/öznitelik katalogunun olmamasıdır. Örneğin [145] de, "mobil telefon baz istasyonu" için bir işaret bulunmamaktadır. UKVA kurumlarının kendi detay/özniteliklerini tanımlamaları ile kartografik gösterim katalogunun ayrıntı düzeyi de belli olacak ve buna göre işaretler belirlenebilecektir. Mevcut kartografik gösterim katalogunun diğer bir yetersizliği, işaretlerin bazen detayı yeterince ifade edememeleridir. Örneğin [145] de telefon santrali ve telefon kulübesi işaretleri yeterince açıklayıcı değildir. Bu yetersizliğin giderilmesi için de, diğer ülkelerde kullanılan tekniklere başvurulabilir.

5. SONUÇLAR

Günümüz beklenti ve sorunlarının birbiriyle ilişkili ve bütünleşik doğası karşısında hızlı, sağlıklı, ve ekonomik çözümler için, konumsal verinin etkin bir biçimde paylaşılması kaçınılmazdır. Ulusal Konumsal Veri Altyapısı (UKVA) ile hedeflenen, ülke düzeyinde konumsal veri erişim ve kullanımını önemli ölçüde iyileştirerek, etkin konumsal veri paylaşımını olanaklı kılacak "bilgi otobanları" dır.

CBS'lere geçiş sürecinin henüz çok başlarında ve bu alandaki öncü ülkelere göre oldukça geri kalmış durumdaki Türkiye'de, konumsal veri ile çalışan kurumların aynı veriyi paylaşmak yerine, ilk elden toplama yoluna giderek kaynaklarını gereksizce harcamaları, geleneksel bir rahatsızlıktır. Kanada ve A.B.D. gibi, CBS ler alanında öncü ülkeler, veri paylaşımı alanında çok geniş bir deneyime sahiptir. Bu ülkelerde çeşitli veri paylaşım programları, 1970 ve 1980 li yıllarda uygulamaya konmuş ve bu programlarda değişen oranlarda başarı sağlanmıştır. Türkiye, CBS'ler alanındaki geri kalmışlığını, bu ülkelerin deneyimlerinden yararlanarak giderme şansına sahiptir. O nedenle Türkiye'nin, veri paylaşımı alanında anılan ülkeler düzeyine erişebilmesi için yapması gereken, onların geçtiği yolları aynen izlemek yerine, kendi UKVA'sını bir an önce kurmaktır.

UKVA, yerel ve Ulusal düzeylerde çeşitli kamu kurumlarına, yerel yönetimlere, özel sektöre, Üniversitelere, ve çeşitli kullanıcı gruplarına hizmet sunacaktır. UKVA ile, çeşitli kullanıcı gruplarının bilgi sunacağı ve/veya bilgi alacağı bir "Bilgi Pazarı" oluşturulacaktır. UKVA ülke genelinde, konumsal veri erişimi ve kullanımını çok önemli ölçüde iyileştirerek, verinin etkin paylaşımını sağlayacak, ve konumsal veriye ulaşmada yaşanan kargaşa ve kaynak israfını önleyerek, ülke ekonomisi için çok önemli bir katkı sağlayacaktır. UKVA çerçevesinde ayrıca, konumsal veri ile çalışan bazı kamu kurumlarının, "Bilgi Çağı" nın gerektirdiği dinamizmi yakalamaları yolunda çoktandır gereksinim duyulan "yeniden yapılanma" sı gerçekleştirilebilecektir. Çünkü UKVA, söz konusu kurumların işleyişi ve yapısı ile ilgili bir takım yasal düzenlemeleri gerektirecektir.

Türkiye Bilgi Çağını yakalamak ve bir Bilgi Toplumu olmak istiyorsa, UKVA'yı bir an önce hayata geçirmelidir. Çünkü UKVA, bütün veri türlerine yönelik olarak oluşturulacak bir Ulusal Veri Altyapısı (UVA)'nın da çok önemli bir bileşeni olacaktır.

UKVA'nın kurulması ve yaşatılması için bir dizi politik, yönetsel, ve teknik gereksinimin karşılanması gerekir. Politik gereksinimler, UKVA'nın kurulması ve yaşatılması ile ilgili temel politikaların belirlenmesine yöneliktir. Yönetsel gereksinimler, UKVA'nın gerek hayata geçirilmesi ve gerekse yaşatılma sürecindeki yönetimi ile ilgilidir. Teknik gereksinimler ise, UKVA'nın gerektirdiği teknik altyapının oluşturulması ve bakımı içindir.

Bu tezde, UKVA'da farklı kullanıcı gereksinimlerini karşılayacak ve taraflar arasında etkin veri paylaşımına olanak tanıyacak bir veri değişim standardı önerilmiştir. Böylece, UKVA için ilk adım atılmış ve çok önemli bir teknik gereksinim karşılanmış olmaktadır.

UKVA konumsal veri değişim standardının belirlenmesi için, Kanada'nın SAIF, A.B.D.'nin SDTS, ve NATO'nun DIGEST formatları, bu tezde belirlenen değerlendirme ölçütlerine göre değerlendirilmiştir. Değerlendirme ölçütleri ile aranan, veri modeli açısından genellik ve genişletilebilirlik, gerçekleştirim açısından esneklik, donanım, ve iletişim ortamı bağımsızlığı ile, değişim ortamı açısından çeviricilerin kolay ve hızlı bir şekilde geliştirilmesine ve etkin bir veri değişimine olanak tanıyacak servis ve araçlar ya da bunları içeren bir Grafik Kullanıcı Arabirimi dir.

Değerlendirme sonucunda SAIF, UKVA gereksinimlerini karşılayacak ve etkin bir veri değişimine olanak tanıyacak en uygun format olarak bulunmuştur. Veri modeli açısından daha kapsamlı ve Nesneye-Dayalı veri modelini kullanması nedeniyle de, daha genişletilebilir bir model sunması bakımından SAIF, SDTS ve DIGEST'den daha üstündür. SAIF'in yüksek bir genişletilebilirlik sunmasının, UKVA açısından önemi çok büyüktür. Çünkü UKVA'da çok farklı kullanıcı grupları olacak ve zamanla yenileri eklenebilecektir. Gerçekleştirim açısından SAIF, detay kodlama bazında yeni bir yaklaşım sunmakta ve SDTS ve DIGEST'in kullandığı alışlagelmiş detay kodlama yerine, farklı kullanıcılara detaylarını istedikleri gibi tanımlama olanağı sunmaktadır. Böylece, çok zor bir iş olan standart detay kodlama kataloglarının oluşturulması ve daha da önemlisi yaşatılmasındaki yük, önemli ölçüde hafifletilmiş olmaktadır. Değişim ortamı açısından da SAIF, çevirici geliştiriciler için DIGEST ve SDTS'e göre çok daha geniş olanaklar sunmaktadır.

Bu sonuca göre, UKVA tarafları veri transfer işlemlerini SAIF formatı üzerinden gerçekleştireceklerdir. Tarafların tek bir formatta veri üretmeleri, UKVA'da konumsal veri değişimini çok önemli ölçüde kolaylaştıracaktır. UKVA'da SAIF'in kullanılması, SDTS ve DIGEST'i geçersiz formatlar yapmayacaktır. SAIF, DIGEST ile doğrudan DIGEST profili üzerinden, SDTS ile de dolaylı olarak SDTS'nin DIGEST profili üzerinden uyumludur.

Dolayısıyla DIGEST ve SDTS formatlarında gelen veri transfer edilebilir ve bu formatlarda veri üretilebilir. Böylece, UKVA'nın uluslararası uyumluluğu sağlanmış olmaktadır.

Çeşitli Uluslararası organizasyonların, konumsal veri yönetiminde standardı sağlama çalışmaları sürmektedir. Bu çalışmalar, veri tanımlamaları ve gösterimleri, yazılım ve donanımları farklı sistemlerin veri paylaşımını, bu tezde değerlendirilen formatlarından daha geniş bir kapsamda, interoperabilite açısından ele almaktadır. Ancak, bu çalışmaların nasıl sonuçlanacağı ve elde edilen sonuçların ne kadar kabul göreceğini kestirmek güçtür; ISO TC 211 oldukça yeni bir organizasyondur ve çalışmalarını sonuçlandırması uzunca bir süreç alabilir. CEN TC 287 ve OGC'nin çalışmalarında çeşitli gelişmeler yaşanmakla birlikte, tamamlanma süreleri belli değildir. Kaldı ki, bu çalışmaların üçünün de ayrı ayrı başarılı olması, üç yeni standart anlamına gelir. Bu durumda hangisi kullanılacaktır? En iyi olasılıkla, bu çalışmalardan tek bir interoperabilite standardının ortaya çıkması ve UKVA'da bu standarda geçilmesi durumunda önemli olan, bunun bu tezde elde edilen sonucu kullanacak UKVA'da yapacağı etkinin minimum olmasıdır. Bu açıdan da en uygun olan format yine SAIF'tir. Çünkü interoperabilite açısından DIGEST ve SDTS'ten daha uygun olan SAIF tir. Başarılı senaryo sonunda elde edilecek interoperabilite standardının SAIF yerine, her üç standardı da kullanması durumunda ise, söz konusu geçişin yapacağı etkinin, UKVA'da SDTS ve DIGEST kullanılması durumundakinden daha olumsuz olacağı da söylenemez. Dolayısıyla, UKVA için SAIF'in önerilmesi, standartlar alanındaki uluslararası gelişmeler açısından da uygundur.

UKVA için SAIF'in önerilmesi ile UKVA'da konumsal veri değişimi problemi yalnızca kısmen çözümlenmiş olmaktadır. Problemin tamamıyla çözümü için, seçilen standarda Türkiye'ye özel bir takım eklemeler yapılması, standart detay/öznitelik katalogları ve işaret kitaplıklarının geliştirilmesi gibi, daha bir dizi gereksinim karşılanmalıdır.

SAIF'e yapılması gerekli başlıca genişletme koordinat sistemleri içindir. SAIF'te tanımlı yatay ve düşey referans ve prjeksiyon sistemlerine, Türkiye'de kullanılan sistemleri eklemek gerekir. SAIF, Türkiye'ye özel koordinat sistemlerinin SAIF standart şemasına eklenmesi için gerekli desteği sunmaktadır. O nedenle burada önemli bir sorun yoktur. Yapılması gereken, SAIF standart şemasında desteklenen ve Türkiye'de kullanılan sistemlerin karşılaştırılarak, varsa gerekli parametrelerin eklenmesidir.

SAIF tasarım ilkesi, farklı kurumların kullandıkları detayları gereksinimlerine en uygun bir biçimde tanımlama esnekliğini sunmaktadır. Gerçekleştirmede de alışlagelmiş detay kodlama yaklaşımı kullanılmamaktadır. O nedenle, her kurumun üzerinde anlaşacağı ortak detay tanımlamalarına ve detay kodlarına gerek yoktur. Ancak, UKVA'da yer alacak

tarafların kullandıkları detay/özniteliklerin tanımlanarak detay kataloglarının oluşturulması gerekir. Burada söz konusu olan yalnızca farklı kurumların kendi detay tanımlamalarını belirlemeleridir. UKVA kurumlarının detay/öznitelik tanımlamaları UKVA "bilgi sunucuları" için de gerekli olacaktır.

UKVA kapsamında ayrıca, mevcut kartografik gösterim katalogunun gözden geçirilerek yeni işaret kitaplıklarının oluşturulması gerekir. UKVA'da standart bir kartografik gösteriminin kullanılması çok önemli bir kolaylık sağlayacaktır. Bunun için, bütün kullanıcı gruplarının üzerinde anlaşacakları ortak bir kartografik gösterim katalogunun oluşturulması gerekir. Türkiye'de bugün kullanılmakta olan kartografik gösterim katalogu yetersiz kalmaktadır. UKVA kurumlarının kendi detay/özniteliklerini tanımlamaları ile, kartografik gösterim katalogu ayrıntı düzeyi de belli olacak ve buna göre işaretler belirlenebilecektir.



6. ÖNERİLER

UKVA için, siyasi otoritenin ve UKVA katılımcılarının politik ve finansal desteklerinin kazanılması gerekir. UKVA siyasi otoriteye, UKVA taraflarına, ve kamuoyuna anlatılarak, ulusal bir politika olarak benimsenmesi sağlanmalıdır. UKVA öncelikle siyasi otorite tarafından desteklenmelidir. Ardından, UKVA taraflarının politik ve finansal desteklerinin kazanılması gerekir. Bunun için kamu kurumları, yerel yönetimler, özel sektör, ve üniversitelerin UKVA'yı anlamaları ve benimsemeleri sağlanmalıdır. Tarafların destek ve bağlılıkları başlangıçta çok güçlü, UKVA'nın yaşatılabilmesi için de sürekli olmalıdır.

UKVA'nın hayata geçirilmesi ve yürütülmesi için yeni bir kurumsal yapılanmaya gereksinim vardır. Tarafların politik ve finansal desteklerinin kazanılması, gerekli koordinasyonun sağlanması, gereksinimlerin karşılanması ve karşılaşılan sorunların giderilmesi bu kurumsal yapılanmanın temel görevleri olacaktır. Bu kurumsal yapılanma bir lider kurum, ve ona bağlı çalışma grupları şeklinde olabilir. Lider kurum UKVA'da yer alacak kamu kurumlarından biri olabilir. Bu kurumun yeterli yaptırım gücü olmalıdır. Bu bakımdan kurumun bir bakanlığa bağlanması düşünülebilir. Ancak, UKVA'nın siyasi otorite değişikliklerinden etkilenmemesi ve kesintiye uğramaması için gerekli önlemler alınmalıdır. Gerek söz konusu kurumsal yapılanma ve gerekse UKVA'nın diğer boyutları için, başta A.B.D., Avustralya, ve Kanada olmak üzere, diğer ülkelerdeki UKVA çalışmalarının gözden geçirilmesi çok yararlı olacaktır.

UKVA'nın bir an önce hayata geçirilebilmesi için, çalışmalar eş zamanlı olarak yürütülmelidir. Bir taraftan siyasi otorite gerekli politikaları belirlerken, diğer yandan UKVA'yı üstlenen kurum, yönetsel ve teknik gereksinimleri, belirlenecek öncelik sırasında ele almalıdır.

HGK'nın "Ulusal CBS projesi" ve Ulusal Coğrafi Veri Tabanı oluşturma çalışmaları ile, DİE'nin genel bir detay/öznitelik kataloğu oluşturma çalışmaları, UKVA kapsamında ve bu tezde elde edilen sonuçlar ışığında yeniden ele alınmalıdır. UKVA çatısı altında, kurumlararası görüş birliğinin sağlanmasında karşılaşılan güçlükler daha kolaylıkla aşılabilecektir.

UKVA için detay/öznitelik kataloglarının hazırlanmasında, FACC'ın detay sınıflaması yaklaşımından yararlanılabilir. FACC uyumlu detay/öznitelik tanımlamaları, UKVA'da DIGEST formatında veri transferi açısından kolaylık sağlayacaktır. Bu çalışmaların da, HGK ve DİE'nin ilgili çalışmaları ile birlikte ele alınması yararlı olacaktır.

Son olarak, UKVA için mevcut kartografik gösterim katalogunun gözden geçirilerek, yeni işaret kitaplıklarının oluşturulmasında, diğer ülkelerde uygulanan işaretlendirme tekniklerine başvurulabilir.



7. KAYNAKLAR

- [1] Forrest, D. Implementing a provincial land information network, Networking Spatial Information Systems, P.W. Newton, P.R. Zwart, M.E. Cavill (Ed.), Belhaven Press, London, 1992, 111-122.
- [2] Marble, D.F., Geographic Information Systems : An Overview, Introductory Readings in Geographic Information Systems, Pequet, D.J., Marble, D.F. (Eds.), Taylor & Francis, 1990.
- [3] Masser, I. ve Blakemore, M., Handling Geographical Information: Methodology and Potential Applications, Longman, UK, 1991.
- [4] Taylor, D.R.F., Geographic Information Systems: The Microcomputer and Modern Cartography, Geographic Information Systems The Microcomputer and Modern Cartography, Taylor, D.R.F. (Ed.), Pergamon press, 1991.
- [5] Maguire, D.J., An Overview and Definition of GIS, Geographical Information Systems: Principles and Applications, Maguire, D. J., Goodchild, M. F., Rhind, D. W. (Eds.), 1991, Longman, London, 515-530...
- [6] Lee, Y.C.ve Zhang, G.Y., Developments of Geographic Information Systems Technology, Journal of Surveying Engineering, 115, 3 (1989) 304-323.
- [7] Feuchtwanger, M., Towards a Geographic Semantic Database Model, Doktora tezi, Simon Fraser University, Ağustos, 1993.
- [8] Dickinson, H. ve Calkins, H.W., The economic evaluation of implementing a GIS, International Journal of Geographic Information Systems, 2, 4 (1988) 307-327.
- [9] Matthews, A.E., Looking at GIS exchange standarts, Mapping Awareness & GIS in Europe, 6, 2 (1992).
- [10] Frank, A., Telecommunication and GIS: Opportunities and Challenges, Networking Spatial Information Systems, P.W. Newton, P.R. Zwart, M.E. Cavill (Editörler), Belhaven Press, London, 1992, 235-249.
- [11] Evangelatos, T.V., Digital geographic interchange standarts, GISs The Microcoputer and Modern Cartography, D.R.F. Taylor (Ed.), Pergamon Press, (1991).
- [12] Lee, Y.C. ve Coleman, D.J., A Framework for evaluating Interchange Standards, CISM (Canadian Institute on Surveying and Mapping) Journal, 44, 4 (1990). 391-402.
- [13] Crosswell, P.L. ve Ahner, A., Computing standarts and GIS: A tutorial, Annual Meeting of the Urban and Regional Information Systems Association (URISA) Proceedings, 1990, Cilt II, 88-105.
- [14] Pascoe, R.T. ve Penny, J.P., Construction of interfaces for the exchange of geographic data, International Journal of Geographical Information Systems, 4, 2 (1990) 147-156.

- [15] Herring, J.R. ve Pammett, K.G. (1992). A corporate database approach to GIS: Techniques for integrating multiple GIS applications, GIS'92 Symposium Proceedings, 1992, Vancouver, B.C, 1-7.
- [16] Kottman, C., A., Some questions and answers about digital geographic information exchange standarts, 2nd Edition, Intergraph Corporation, Huntsville, Alabama, US, 1992.
- [17] Coleman, D.J. ve McLaughlin J. D., Standards for spatial information interchange: A management perspective, CISM (Canadian Institute on Surveying and Mapping) Journal, 46, 2 (1992) 133-141.
- [18] McLaughlin J. ve Nichols, S, Developing a National Spatial Data Infrastructure, Journal of Surveying Engineering, 120, 2 (1994) 62-76.
- [19] MSC (Mapping Science Committee), Toward a Coordinated Spatial Data Infrastructure for the Nation, National Academy Press, Washington, D.C, 1993.
- [20] Tycon, G.G. ve Johnson, M.R., GIS Data Exchange: Standards and Formats, GIS'90 Symposium, Vencouver, British Columbia, Mart 1990, 269-276.
- [21] Moellering, H., Approaches to Spatial Database Standards: An Introduction, Spatial Database Transfer Standards: Current International Status, Moellering, H. (Ed.), International Cartographic Association (ICA), Elsevier, 1991.
- [22] Fegeas, R.G., Cascio, J.L. ve Lazar, R.A., An Oview of FIPS 173, The Spatial Data Transfer Standard, Cartography and Geographic Information Systems, 19, 5 (1992) 1-24.
- [23] O'Brien C.D. , Evangelatos, T. ve McKellar, D., The Harmonization of Geomatics Standards, Canadian Conference on GIS Proceedings, 83-92, Ottawa, 1994.
- [24] Province of British Columbia, Lands and Parks, Report on Land-Related Data Sharing Developments in British Columbia, 1994.
- [25] EDI and Joint Use of Geographic Information, Workshop 3, 17th Urban Data Management Symposium, Helsinki-Espoo, Finland, Eylül 1994.
- [26] Ulusal Enformasyon Altyapısı, TÜBİTAK Bilim ve Teknik Dergisi, Eylül 1995, 26-30.
- [27] IHO (International Hydrographic Organization), DX90: Specification for the exchange of digital hydrographic data, working draft, Committee on Exchange of Digital Data, 1990.
- [28] Buchanan, H.J., BS7567 (NTF) and the emerging European (CEN) transfer standard, International Symposium on GIS Standards, Seoul, Korea, Kasım 1995, H1-H9.
- [29] CEN (European Committee for Standardization) Report, Geographic Information-Reference Model-Working Draft-Version 2., CEN/TC 287 N 154, 1993.
- [30] Buehler, K.A., The OGIS Technical Committee: Purpose, Status, and Future, Geo Info Systems, Ocak (1995) 48.
- [31] OGC (Open GIS Consortium), OGIS Technical Summary, <http://www.ogis.org>, 1996.

- [32] GIS World Inc., GIS World Outlook'96, Software and Operating Systems, Aralık (1995) 40-49.
- [33] International Standards Organization (ISO)/TC 211/Ad hoc WG 1, Geographic Information Standards Reference Model, Mart 1995.
- [34] Aydemir, S., Bank, E. ve Taştan, H., Harita Genel Komutanlığında Yürütülen Coğrafi Bilgi Sistemi Projesi, Uluslararası Bilgisayar Uygulamaları Sempozyumu, Haziran 1993, Konya, 209-222.
- [35] Surveys and Resource Mapping Branch, Ministry of Environment, Lands and Parks, British Columbia (BC), Spatial Archive and Interchange Format: Formal Definition, Reference Series Vol.1, Release 3.1, Nisan 1994.
- [36] Elmasri, R. ve Navathe, S.B., Fundamentals of Database Systems, Benjamin/Cummings, California, 1989.
- [37] Masry, S.E. ve Lee, Y.C., An Introduction to Digital Mapping, Department of Surveying Engineering publication, UNB, Canada, 1988.
- [38] Wertz, C.J. The Data Dictionary Concepts and Uses, Second Edition, QED Information Sciences Inc., Massachusetts, 1989.
- [39] Lee, Y.C. ve Isdale, M., The Need for a Spatial Data model, Canadian Conference on GIS, 1991, Ottawa, 530-531.
- [40] Surveys and Resource Mapping Branch, Ministry of Environment, Lands and Parks, BC, Spatial Archive and Interchange Format: Overview of SAIF, Reference Series Vol.1, Release 1.0, Ocak 1991.
- [41] Tsichritzis, D. ve Klug, A., (Eds.) The ANSI/X3/SPARC DBMS Framework, AFIPS Press, 1978.
- [42] Herring, J. R., TIGRIS: Topologically Integrated Geographic Information System, Eight International Symposium on Computer-Assisted Cartography (Auto Carto 8) Proceedings, 1987, Washington, DC.
- [43] Worboys, M.F. ve Deen, S.M., Semantic Heterogeneity in Distributed Geographic Databases, ACM SIGMOD Record, 20, 4 (1991) 30-34.
- [44] Morehouse, S., Arc/Info - A Geo-Relational Model for Spatial Information, Seventh International Symposium on Computer-Assisted Cartography (Auto Carto 7) Proceedings, 1985, Washington, DC, 388-397.
- [45] Samet, H., The design and Analysis of Spatial Data Structures. Addison-Wesley, 1990.
- [46] Robson, R.N., Database Management Systems II. Lecture Notes, Univesity of New Brunswick, Canada, 1988.
- [47] Codd, E., A Relational Model for Large Databanks, Communications of ACM, 13, 6 (1970).
- [48] Kent, W. A simple guide to five normal forms in relational database theory. Communications of the ACM, 26, 2 (1983) 120-125.

- [49] Codd, E.F., Further normalization of the database relational model. Data base Systems, Rustin, R. (Ed.), Prentice Hall, 1972.
- [50] Neely, J. ve Stewart, S., Fundamentals of Relational Data Organization, BYTE Publications Inc., Kasım 1981, 48-60.
- [51] Codd, E.F., Extending the database relational model to capture more meaning, ACM TODS, 4, 4 (1979) 397-434.
- [52] Abel, D.J., SIRO-DBMS: A Database Tool-kit for Geographical Information Systems, International Journal of Geographical Information Systems, 3, 2 (1989) 103-116.
- [53] Haas, L.M. ve Cody, W.F., Exploiting extensible DBMS in Integrated Geographic Information Systems, Technical Report RJ 8173 (75132), IBM, 1991.
- [54] Smith, K.E. ve Zdonik, S.B. Intermedia: A case study of the differences between relational and object-oriented database systems. OOPSLA proceedings, 1987, 452-465.
- [55] Stonebraker, M. ve Rowe, L.A., The Design of POSTGRES, ACM SIGMOD International Conference on the Management of Data Proceedings, 1986, Washington D.C, 340-355.
- [56] Haskin, R.L. ve Lorie, R.A. On Extending the Functions of a Relational Database System, ACM SIGMOD Conference Proceedings, 1982, 207-212.
- [57] Potter, D.W. ve Trueblood, R.P., Traditional, Semantic, Hyper-Semantic Approaches to data modeling, IEEE Computer, Haziran 1988, 53-63.
- [58] Hammer, M. ve McLeod, D., Database Description with SDM: Semantic Data Model, ACM Transactions on Database Systems, 6, 3 (1981), 25-43.
- [59] Shipman, D. The Functional Data Model and the Data language DAPLEX, ACM Transactions on Database Systems, 6, 1, (1981).
- [60] Kim, W., Object-Oriented Databases: Definition and Research Directions, IEEE Transactions on Knowledge and Data Engineering, 2, 3 (1990).
- [61] Kemp, Z. ve Oxborrow, E., An Object Model for Distributed Multimedia Geographic Data, European GIS Conference, Mart 1992, Münih, 1294-1303.
- [62] Chen, P., The Entity Relationship Model—Toward a Unified View of Data, ACM Transactions on Database Systems, 1, 1 (1976) 9-36.
- [63] Smith, J. ve Smith, D., Database Abstractions: Aggregation and Generalization, ACM Transactions on Database Systems, 2, 2 (1977).
- [64] Meyer, B., Object-Oriented Software Construction, Prentice Hall International, Hemel Hempstead, UK, 1988.
- [65] Khoshafian, S. ve Abnous, R., Object-Orientation: concepts, language, databases, user interfaces. John Willey & Sons, New York, 1990.
- [66] Hughes, J.G., Object-Oriented Databases, Prentice Hall, London, UK, 1991.

- [67] Brookshear, J.G., Computer Science, Third Edition, Benjamin/Cummings, California, US, 1991.
- [68] Wiederhold, G. Database Design, 2nd Edition, McGraw-Hill, Singapur, 1983.
- [69] Voelcker, J., Helping computers communicate, IEEE Spectrum, Mart (1986) 61-70.
- [70] Pascoe, R.T., Construction of Interfaces for the transfer of Data between Geographical Information Systems, Doktora Tezi, University of Canterbury, New Zeland, 1994.
- [71] ISO (International Standards Organization), Basic Reference Model (ISO 7498), Information Processing Systems-Open Systems Interconnection, 1984.
- [72] Chatterjee, A. ve Segev, A., Data Manipulation in Heterogeneous Databases, ACM SIGMOD Record, 20, 4 (1991) 64-68.
- [73] Sheth, A.P. ve Larson, J.A., Federated Database Systems for Managing, Distributed, Heterogeneous, and Autonomous Databases, ACM Computing Surveys, 22, 3 (1990) 183-235.
- [74] ISO (International Standards Organization), Specification for Abstract Syntax Notation one (ASN.1), Information Processing Systems-Open Systems Interconnection, 1987.
- [75] Pascoe, R.T. ve Penny, J.P., Transforming Geographic Data between different concrete representations, 16th Australian Computer Science Conference Proceedings, Gupta, G., Mohay, G., Topor, R. (Ed.), 1993, 633-653.
- [76] Sondheim, M., Lowe, G., Menes ve R., Friesen, P., A Data Model for Geomatics, GIS for 1990s Conference Proceedings, 1990, Ottawa, 1389-1404.
- [77] Naur, P., Revised Report on the Algorithm Language ALGOL 60, Communications of the ACM, 6, 1 (1963) 1-17.
- [78] ISO (International Standards Organization), Specification for a data descriptive file for information interchange, Information Processing Systems-Open Systems Interconnection, 1985.
- [79] Sondheim, M., Kişisel İletişim, 1994.
- [80] CHS (Canadian Hydrographic Service), Department of Fisheries and Oceans, Specification of the Map and Chart data Interchange Format (MACDIF), Version 2, Ottawa, 1988.
- [81] Lee, Y.C., An Object-Oriented Environment for GIS Data Exchange, GIS for the 1990s Conference Proceedings, Mart 1990, Ottawa.
- [82] Koo, T.K., Graphic database exchange: Direct translators versus universal translators, The Cartographic Journal, 25, 2 (1988) 104-108.
- [83] Guphill, S.C., Spatial data exchange and standardization, Geographical Information Systems: Principles and Applications, Maguire, D. J., Goodchild, M. F., Rhind, D. W. (Eds.), 1991, Longman, London, 515-530.
- [84] Friesen, P., Kucera, H. ve Sondheim, M. SAIF profiles: the missing link, GIS'93 Symposium Proceedings, 1993, Vancouver, B.C., 1113-1120.

- [85] Buchanan, H.J., NTF: An introduction, Mapping Awareness and GIS in Europe, 6, 5 (1992) 38-41.
- [86] Autodesk Inc., AutoCAD Release 12, Customization Manual, Neuchatel, Switzerland, 1992.
- [87] Department of the Interior, USGS (U.S. Geological Survey), National Mapping Division, Standarts for Digital Line Graphs. National Mapping Program Technical Instructions, Part 2, Specifications, Virginia, 1988.
- [88] Zeiler, M.N., Complete Translation of Graphical and Attribute Cartographic Information between Computer File Formats, ASPRS-ACSM Annual Convention, Technical Papers, 1987 Cilt 5, 141-150.
- [90] Lee, Y.C., Kişisel İletişim, 1993.
- [91] Mamrak, S., Kaelbling, M., Nicholas, C.K. ve Share, M., Chamelon: A system for solving the data-translation problem, IEEE Transactions on Software Engineering, 15, 2 (1989) 1090-1108.
- [92] Mak, R., Writing Compilers and Interpreters. An Applied Approach, Willey & Sons, New York, 1991.
- [93] Lee, Y.C., Kişisel İletişim, 1992.
- [94] Calkins, H.W., Institutions sharing spatial information, Networking Spatial Information Systems, P.W. Newton, P.R. Zwart, M.E. Cavill (Ed.), Belhaven Press, London, 1992, 283-292.
- [95] Coleman, D.J. ve Ogilvie, M., Reaching Consensus on Spatial Data Standards at the State/Provincial Level: A Canadian Example, URISA'91 Conference, Kasım 1991, 457-466.
- [96] Fenten, G. Coordination of the area of geographical information in the Netherlands, Joint European Conference and Exhibition on Geographical Information, Mart 1995, The Hague, Netherlands, 12-15.
- [97] Montgomery, G.E ve Schuch, H.C., GIS Data Conversion Handbook, GIS World Inc., Colerado, USA, 1993.
- [98] Coleman, D.J. ve McLaughlin J.D, Building a Global Spatial Data Infrastructure: Usage Paradigms and Market Influences, Geomatica, 48, 3, (1994) 225-236.
- [99] Crosswell, P., Using wide area communication networks to support GIS, Geo Info Systems, 5,1 (1995) 40-42.
- [100] Cömert, Ç., Nesneye-Yönelim Tekniğinin Coğrafi Bilgi Sistemleri için bir değerlendirmesi, 1. Ulusal Coğrafi Bilgi Sistemleri Sempozyumu, 18-20 Ekim, 1994, Trabzon, 65-79.
- [101] Lee, Y.C. ve McLaughlin J., Distributed Land Information Networks: Database management issues , CISM (Canadian Institute on Surveying and Mapping) Journal, 45, 3 (1991) 353-363.

- [102] Orlowska, M.E., Lister, A.M. ve Fogg, I. Decentralising spatial databases, Networking Spatial Information Systems, P.W. Newton, P.R. Zwart, M.E. Cavill (Ed.), Belhaven Press, London, 1992, 63-76.
- [103] Cerf, V.G., Networks, Scientific American, 265, 3 (1991), 72-81.
- [104] Newton, P.V., Zwart, P.R. ve Cavill, M.E., Inhibitors and Facilitators in High Speed Networking of Spatial Information Systems, Networking Spatial Information Systems, P.W. Newton, P.R. Zwart, M.E. Cavill (Eds.), Belhaven Press, London, 1992, 251-263.
- [105] Burrough, P. A., Principles of Geographical Information Systems for Land Resources Assessment, Clarendon Press, Oxford, 1986.
- [106] Ehlers, M., Greenlee, D., Smith, T. ve Starr J., Integration of remote sensing and GIS: Data and data access, Photogrammetric Engineering & Remote Sensing, 5, 6 (1991) 669-675.
- [107] Guptil, S. ve Starr, L.E., The Future of Cartography in the Information Age, Computer Assisted Cartography Research and Development Report, ICA (International Cartographic Association), Washington, 1984.
- [108] West, B.H., Griesbach, E.H., Taylor, J.D. ve Taylor, L.T., The Prentice-Hall Encyclopaedia of Mathematics, Prentice-Hall, New Jersey, 1982.
- [109] Gatrell, A.C., Concepts of Space and Geographical Data, Geographical Information Systems: Principles and Applications, Maguire, D.J., Goodchild, M.F., Rhind, D.W. (Ed.), Longman, London, 119-134, 1991.
- [110] Feuchtwanger, M., Geographic Logical Database Model Requirements, Auto-Carto 9 Proceedings, 1989, Baltimore, Maryland.
- [111] Ralphs, M., Data Quality Issues and GIS- A discussion, Mapping Awareness & GIS in Europe, Eylül (1993).
- [112] Ventura, S., Standards offer LIS/GIS support, GIS World, Ağustos (1993) 48-50.
- [113] Price, S., Modelling the Temporal Element in Land Information Systems, International Journal of Geographical Information Systems, 3, 3 (1989) 233-243.
- [114] Aangeenbrug, R.T., A Critique of GIS, Geographical Information Systems: Principles and Applications, Maguire, D.J., Goodchild, M.F., Rhind, D.W. (ed.), Longman, London, UK, 1991, 101-107.
- [115] Edwards, G., Gagnon, P. ve Bedard, Y., Spatial-Temporal Topology and Casual Mechanisms in time Integrated GS: From Conceptual Model to implementation Strategies, Canadian Conference on GIS, 1993, Ottawa, 842-857.
- [116] Barrera R., Frank A. ve Al-Taha K., Temporal Relations in Geographic Information Systems, ACM SIGMOD Record, 20, 3 (1991) 85-91.
- [117] Vrana, R., Historical Data as an Explicit Component of Land Information Systems, International Journal of Geographical Information Systems, 3, 1 (1989) 33-49.
- [118] Pequet, D.J. ve Marble, D.F., GIS Internals-Data Representation and Analysis Techniques, Introductory Readings in Geographic Information Systems, Pequet, D.J., Marble, D.F. (Ed.), 1990, 247-249.

- [119] Nyerges, T.L., Schema Integration Analysis for the Development of GIS Databases, International Journal of Geographical Information Systems, 3, 2 (1989) 153-183.
- [120] Menes, R ve Sondheim, M., A chicken in every pot...and a GIS on every desk, GIS'90 Symposium Proceedings, Mart 1990, Vancouver, B.C., 247-252.
- [121] Altheide, F., An Implementation Strategy for SDTS Encoding, Cartography and Geographic Information Systems, 19(5) (1992) 306-310.
- [122] Owen, J., Data Interchange? A Personal View of Today's Technology, Fourth European GIS Conference Proceedings (EGIS), 1993, Genoa, Italy, 25-29.
- [123] NATO, Feature and Attribute Coding Catalogue (FACC), Unclassified Publication, 1993.
- [124] Hermansen., C., An Evaluation of DIGEST, Master Tezi, University of New Brunswick, Fredericton, 1994.
- [125] Interactive Software Engineering Inc., Eiffel: The Environment, TR-EI-5/UM Version 2.3, California, US, 1990.
- [126] Marx, R.W., The TIGER system: Automating the geographic structure of the United States Census, Government Publications Review, Cilt 13, 1986, 181-201, Pergamon Press.
- [127] GIS World, Inc., International GIS Sourcebook, Fort Collins, Colorado, 1995.
- [128] SAIF Frequently Asked Questions, infosafe@safe.com, Eylül, 1995.
- [129] Surveys and Resource Mapping Branch, Ministry of Environment, Lands and Parks, BC, Spatial Archive and Interchange Format: Formal Definition, Reference Series Vol.1, Release 2.0, Ağustos 1992.
- [130] Sondheim, M., Kişisel İletişim, 1993.
- [131] Safe Software Inc., Spatial Archive and Interchange Format: Encoding Schemes, SAIF/ZIP Data Set File Format, Release 1.3, BC, Haziran 1994.
- [132] Safe Software Inc., SAIF Toolkit API. Programmer's Reference Manual, Release 0.9, 1994.
- [133] Kucera, H.A., Chin, R.S. ve Jameson, C.L., SAIF - Conceptualization to Realization, GIS'93 Symposium, Vancouver, BC, Şubat 1993, 1121-1129.
- [134] Lutz, D., Kişisel İletişim, 1994.
- [135] SDTS Frequently Asked Questions, @sdts.er.usgs.gov, Aralık, 1995.
- [136] Milne, P., Scott, M. ve Smith, J.L., Geographical Object-Oriented Databases—A Case Study, International Journal of Geographical Information Systems, 7, 1 (1993) 39-55.
- [137] Moore, L., Kişisel iletişim, 1995.
- [138] Strand, E.J., ISO Extends Structured Query Language the "SAIF" Way, GIS World, Ocak (1994) 24-25.

- [139] Lazar, R.A., SDTS Support Software: The FIPS 123 Function Library, Cartography and Geographic Information Systems, 19, 5 (1992) 303-305.
- [140] George, R. The Spatial Data Transfer Standard Faces Challenges, 1991-92 International GIS Sourcebook (1992) 425-426.
- [141] GIS-L tartışmaları, GIS-L tartışma listesi (comp.infosystems.gis), 1993.
- [142] McKellar, D., Geographic Information: The Way Ahead, Canadian Conference on GIS Proceedings, 1994, Ottawa, 1699-1706.
- [143] Phillips, M. , The Digital Geographic Information Exchange Standards (DIGEST), Canadian Conference on GIS Proceedings, 259-270, Ottawa, 1993.
- [144] NATO, Digital Geographic Information Exchange Standard (DIGEST), AGeoP-3, Vol I-II, 1st draft, Unclassified Publication, 1993.
- [145] TMMOB Harita ve Kadastro Mühendisleri Odası-İstanbul Şubesi, Büyük Ölçekli Haritaların Yapım Yönetmeliği, İstanbul, 1989.



8. EKLER

Tezde kullanılan teknik terimlerin İngilizce karşılıkları :

Adresleme: Hash/Hashing

Ağ model: Network model

Ağaç: Tree

Alt-sistem: Sub-system

Amaca özel (anlık): ad-hoc

Anahtar: Key

Anlam: Semantics

Arabirim: Interface

Araç: Tool

Ayrıştırıcı: Parser

Ayrıştırma: Decomposition

Bağlaç: Link

Bilgi Sunucuları: Information Servers

Bilgi Tabanlı Sistemler: Knowledge Based Systems

Bilgisayar Destekli Tasarım/Üretim: Computer Aided Design/Production

Birleştirme: Aggregation

Bütünleşik sistemler: Integrated Systems

Bütünleşik: Integrated

Bütünleşme: Integration

Coğrafi Bilgi Sistemleri (CBS): Geographical Information Systems

Coğrafi nesne: Geographic object

Çeviri: Translation

Çevrim içi: On-line

Çevrim-dışı: Off-line

Çok düzeyli indeksler: Multi level indexes

Çözünürlük: Resolution

Değişim standardı: Interchange standard

Değişken-uzunluklu: Variable-length

Değişken: Variable

Depolama: Storage

Detay kodu: Feature code

Detay: Feature
Dizi: Array
Domain: Domain
Dosya tanımlayıcısı: File descriptor (header)
Dörtlü-ağaç: Quadtree
Düzenli dosya: Ordered file
Düzensiz dosya: Unordered file
Ebeveyn-çocuk: Parent-child
Entity: Varlık
Etiket: Tag
Format: Format
Genelleştirme: Generalization
Genişletilebilirlik: Extendibility
Genişletme: Extension
Gerçek: Reality, Phenomena, Real world, Miniworld
Gerçekleştirici: Executor
Gerçekleştirim veri modelleri: Implementation data models
Gerçekleştirim: Implementation
Görselleştirme: Visualisation
Görüntü İşleme Sistemleri: Image Processing Systems
Göstergeç: Pointer
Gösterim: Representation
Göze: Cell
Grafik Kullanıcı Arabirimleri: Graphical User Interfaces (GUIs)
Gruplama: Clustering
Hiyerarşik model: Hierarchical model
İkili arama: Binary search
İletişim servisleri: Communication services
İlişkilendirme: Association
İlişkisel cebir: Relational algebra
İlişkisel hesap: Relational calculus
İlişkisel model: Relational model
İndeksleme: Indexing
İnteroperabilite: Interoperability
İşaret: Symbol
Karmaşık varlık: Complex (Composite) entity
Karmaşık varlık: Complex (Composite) entity
Katman-tabanlı: Layer-based

Kavramsal veri modelleri: Conceptual data models
Kavramsal: Conceptual
Kayıt tipi: Record type
Kayıt-tabanlı: Record-based
Kayıt: Record
Kod çözümleme: Decoding
Kod üretici: Code generator
Kodlama: Encoding
Konumsal veri modelleri: Spatial data models
Konumsal veri: Spatial data
Kuyruk: Que
Liste: List
Mimari: Architecture
Modülerlik: Modularity
Nesne paylaşımı: Object sharing
Nesne-tabanlı: Object-based
Nesne: Object
Nesneye Yönelim: Object Orientation
Nesneye-Dayalı veri modeli: Object-Oriented data model
Oluşum: Composition
Özelleştirme: Specialization
Özyineli ayrıştırma: Recursive decomposition
Rastgele Erişim: Random (direct) access
Resmi Tanımlama Dilleri: Formal Description Languages
Sabit-uzunluklu: Fixed-length
Semantik veri modelleme: Semantic data modelling
Set: Set
Seyrek indeks: Non-dense index
Sığ indeks: Dense index
Sınıf hiyerarşisi: Class hierarchy
Sınıf örnekleri: Instances
Sınıf-oluşum hiyerarşisi: Class-composition hierarchy
Sıralı Erişim: Sequential access
Somut sözdizim: Concrete syntax
Soyut sözdizim: Abstract syntax
Soyut Veri Tipleri: Abstract Data Types
Sözdizim ağacı: Syntax tree
Sözdizim: Syntax

Şema: Schema

Tarayıcı: Scanner

Tekrarlı kod kullanımı: Code reuse (Reusability)

Topoloji oluşturma (kurma): Topology building

Varlık-İlişki modeli: Entity-Relationship model

Veri değişim formatı: Data interchange format

Veri değişimi: Data interchange

Veri elemanı: Data item

Veri fazlalığı : Data Redundancy

Veri grubu: Data set

Veri modeli: Data Model

Veri modelleme : Data modelling

Veri Paylaşımı: Data sharing

Veri soyutlama: Data abstraction

Veri sözlüğü: Data dictionary

Veri Tabanı Yönetim Sistemi (VTYS): Database Management System (DBMS)

Veri Tanımlama Dili: Data Description Language

Veri tekrarı: Data duplication

Veri yapıları: Data structures

Versiyon sürekliliği: Version maintenance

Yanıt zamanı: Response time

Yığıt: Stack

Yorumlayıcı: Interpreter

9. ÖZGEÇMİŞ

1963 yılında Trabzon'da doğdu. İlk ve orta öğrenimini aynı şehirde tamamladı. 1981 yılında girdiği, Karadeniz Teknik Üniversitesi Jeodezi ve Fotogrametri Mühendisliği Bölümü Lisans programını 1985 yılında tamamlayarak, aynı yıl Yüksek Lisans öğrenimine ve aynı bölümde Araştırma Görevlisi olarak çalışmaya başladı. 1987 yılında Yüksek Lisans öğrenimini tamamladı. 1988 yılında doktora öğrenimi için, Karadeniz Teknik Üniversitesi tarafından, Kanada'nın New Brunswick Üniversitesi'ne gönderildi. Bu üniversitenin Surveying Engineering bölümündeki çalışmalarının ardından, 1993 yılında Karadeniz Teknik Üniversitesi Jeodezi ve Fotogrametri Mühendisliği Bölümü'ndeki görevine geri döndü ve doktora çalışmalarına devam etti.