

EGE ÜNİVERSİTESİ FEN BİLİMLERİ ENSTİTÜSÜ

(DOKTORA TEZİ)

**COĞRAFİ BİLGİ SİSTEMLERİNİN OWL VE GML
KULLANARAK ANLAMSAK WEB İLE
BÜTÜNLEŞTİRİLMESİ**

Murat KOMESLİ

Bilgisayar Mühendisliğı Anabilim Dalı

Bilim Dalı Kodu: 619.01.00

Sunuş Tarihi: 23.02.2007

Tez Danışmanları: Yrd.Doç.Dr. Murat Osman ÜNALIR

Doç.Dr. Vahap TECİM

Bornova-İZMİR

III

Murat KOMESLİ tarafından DOKTORA TEZİ olarak sunulan “**Coğrafi Bilgi Sistemlerinin OWL ve GML Kullanarak Anlamsal Web ile Bütünleştirilmesi**” başlıklı bu çalışma, E.Ü. Lisansüstü Eğitim ve Öğretim Yönetmeliği ile E.Ü. Fen Bilimleri Enstitüsü Eğitim ve Öğretim Yönergesi’nin ilgili hükümleri uyarınca tarafımızdan değerlendirilerek savunmaya değer bulunmuş ve 23.02.2007 tarihinde yapılan tez savunma sınavında aday, oybirliği/oyçokluğu ile başarılı bulunmuştur.

Jüri Üyeleri :

İmza

Jüri Başkanı : Prof.Dr. Şaban EREN

.....

Raportör Üye: Yrd.Doç.Dr. M.Osman ÜNALIR

.....

Üye : Doç.Dr. Vahap TECİM

.....

Üye : Yrd. Doç.Dr. Adil ALPKOÇAK

.....

Üye : Yrd.Doç. Dr. Vecdi AYTAÇ

.....

ÖZET**COĞRAFI BİLGİ SİSTEMLERİNİN OWL VE GML
KULLANARAK ANLAMSAK WEB İLE
BÜTÜNLEŞTİRİLMESİ**

KOMESLİ, Murat

Doktora Tezi, Bilgisayar Mühendisliği Bölümü

Tez Yöneticileri: Yrd.Doç.Dr. Murat Osman ÜNALIR

Doç.Dr. Vahap TECİM

Şubat 2007, 80 sayfa

Bu tez kapsamında, coğrafi verilerin heterojen yapıdaki coğrafi bilgi sistemleri veritabanları arasında paylaşımını sağlamak maksadıyla, anlamsal web teknolojisi kullanılmaktadır. Bu amaçla, C# programlama dili kullanılarak geliştirilen uygulama yazılımı yardımıyla, bir RDF dosyası içerisindeki coğrafi veriler, veri tabanları arasında değiştirilmektedir. Coğrafi veri hakkında daha detaylı bilgiye erişmek için, web sayfaları da mevcuttur.

Bu amaçla, anlamsal webe dayalı coğrafi bilgi sistemlerinin coğrafi bilgileri anlamaları ve birbirleri arasında değişmelerine hizmet etmek amacıyla OWL dili kullanarak bir coğrafi ontoloji geliştirilmiştir.

Proje kapsamında gerçekleştirilen veri alışverişi temel olarak sistemlerin gerçek kapasitelerini yansıtmamaktadır. Ancak, ilk bütünleştirme adımı olarak kabul edilebilir. Ayrıca, aynı amaca hizmet etmek maksadıyla GML kullanılması önerilmiştir.

Anahtar sözcükler: anlamsal web, ontoloji, coğrafi bilgi sistemleri, coğrafi veri.

ABSTRACT

**INTEGRATION OF GEOGRAPHIC INFORMATION
SYSTEMS WITH SEMANTIC WEB BY USING GML AND OWL**

KOMESLİ, Murat

Ph.D. in Computer Engineering Department

Supervisors: Assist.Prof.Dr. Murat Osman ÜNALIR

Assoc.Prof.Dr. Vahap TECİM

February 2007, 80 pages

Within the context of this thesis, the semantic web technology has been utilized in order to manage to share the geographic data among the heterogeneous geographical information systems' databases. To achieve this aim, geographic data which is stored in an RDF file have been exchanged among the databases by means of an application software, developed using C# programming language. Web pages are also existent to reach the detailed information about the geographic data.

To achieve the aim, a geographic ontology has been developed by using OWL language in order the geographic information systems relying on semantic web to serve to understand and exchange the geographic data between each other.

Basicly, the exchange of data developed in this thesis is not reflecting the real capacity of the systems. But, it can be considered as a first entegration step. Furthermore, with the same purpose, the use of GML has been proposed.

Keywords: semantic web, ontology, geographic information systems, geographic data.

IX

TEŞEKKÜR

Bu tez çalışması süresince, sonsuz ilgi ve sabır ile değerli katkılarını hiç bir zaman esirgemeyen danışmanlarım sayın Yrd.Doç.Dr. Murat Osman ÜNALIR'a ve sayın Doç.Dr. Vahap TECİM'e teşekkürlerimi sunarım.

Ayrıca, tez üzerindeki yapıcı önerilerinden dolayı, başta jüri başkanı sayın Prof.Dr. Şaban EREN olmak üzere saygıdeğer jüri üyelerime teşekkür ederim.

Hiç bir zaman eksilmeyen sevgi ve desteklerinden dolayı sevgili eşim Uzm.Ecz. Yelda KOMESLİ, oğlumuz Kıvanç ve kızımız Ahenk'e minnettar olduğumu belirtmek isterim.

Son olarak, Ege Üniversitesi Bilgisayar Mühendisliği Bölümü'nün değerli öğretim üyesi, elemanları ve idari personeline saygı ve teşekkürlerimi sunarım.

İÇİNDEKİLER

Sayfa

ÖZET.....	V
ABSTRACT.....	VII
TEŞEKKÜR.....	IX
İÇİNDEKİLER	XI
ŞEKİLLER DİZİNİ.....	XIV
KISALTMALAR DİZİNİ	XVI
TEKNİK TERİMLER LİSTESİ.....	XVII
1. GİRİŞ	1
2. COĞRAFİ BİLGİ SİSTEMİ VE COĞRAFİ VERİ.....	4
2.1. Coğrafi Bilgi Sistemi Bileşenleri.....	6
2.2. Grafiksel Veri ve Modellenmesi.....	9
2.3. Coğrafi Bilgi Sistemi İşlevleri.....	12
2.3.1. Veri Toplama.....	12
2.3.2. Veri Depolama.....	13
2.3.3. Veri İşleme ve Dönüşüm.....	14
2.3.4. Veri Analizi.....	14
2.3.5. Veri Sunumu (Gösterim).....	15
2.4. Coğrafi Veri ve Analizler.....	15
2.4.1. Konum Belirleme.....	15
2.4.2. Temel Coğrafi Nesneler.....	20
2.4.3. Temel Ölçümler.....	20
2.4.4. Veri Sorgulama.....	21
2.4.5. Veri Sınıflandırma.....	22

İÇİNDEKİLER (devam)

2.4.6. Tampon Bölge Oluşturma.....	23
2.4.7. Katmanların Çakıştırılması.....	25
2.4.8. Yüzey Analizleri.....	27
2.4.9. Ağ Analizleri.....	30
3. ANLAMSAL WEB VE ONTOLOJİ.....	33
3.1. Anlamsal Web'in Tanımı.....	33
3.2. Anlamsal Web'in Temel Prensipleri.....	35
3.3. Anlamsal Web'in Katmanları.....	38
3.4. Ontoloji.....	39
3.4.1. Ontoloji Özellikleri.....	39
3.4.2. Anlamsal Web ve Ontoloji.....	40
3.4.3. RDF ve RDFS.....	41
4. COĞRAFİ ANLAMSAL WEB (GEO SEMANTIC WEB).....	43
4.1 Coğrafi İşaretleme Dili.....	44
4.2 GML'in Kapsamı.....	47
4.3. Coğrafi Ontoloji Dili Özellikleri.....	48
4.4. DAML+OIL.....	49
4.5. GML ve DAML+OIL.....	51
4.6. Coğrafi Ontoloji.....	52
4.6.1. SWOOP ve Protégé Yazılımlarının Karşılaştırılması.....	52
4.6.2. Topolojik İlişkiler.....	53
5. UYGULAMA.....	58
5.1 Ontoloji ve Olgularının Geliştirilmesi.....	59
5.1.1. Bilgi Değişim Veri Modeli Ontolojisi.....	59
5.1.2. Sorgu ontolojisi.....	59
5.1.3. Coğrafi Ontoloji	64
5.2 Coğrafik Verilerin Uygulamalar Arası Paylaşımı.....	65

XIII

İÇİNDEKİLER (devam)

5.2.1 Aramanın Yapılması.....	67
5.2.2 SOAP Kodlandırması.....	71
6. SONUÇ VE ÖNERİLER.....	78
KAYNAKLAR.....	81
EKLER: EK 1 COĞRAFİ ONTOLOJİ.....	88
ÖZGEÇMİŞ.....	95

ŞEKİLLER DİZİNİ

Sekil

Sayfa

2.1: CBS'nin gerçek dünya ile ilişkilendirilmesi.....	4
2.2: CBS ile diğer Bilgi Sistemleri arasındaki ilişki.....	5
2.3: Coğrafi Bilgi Sistemi Temel Bileşenleri.....	8
2.4: Uzaysal verilerin vektörel olarak gösterimi ve x, y koordinatlarına dönüştürülmesi.....	10
2.5: Grafiksel veri modelleri.....	11
2.6: Bilgisayara bağlanmış bir sayısallaştırma masası.....	12
2.7: Uydu fotoğrafları otomatik veri toplama yöntemlerinden biridir.....	13
2.8: CBS'de verilerin sunumu	16
2.9: Coğrafi Grid Sistemi.....	17
2.10: Harita ölçekleri ve gösterilen detaylar	19
2.11: Verilerin sınıflandırılması.....	23
2.12: Nokta, çizgi ve alan sembolleri için tampon bölge oluşturma örnekleri	24
2.13: İki vektörel katmanın birleştirilmesi.....	27
2.14: Bir bölgenin tel kafes şeklinde çizilmiş yüzey modeli	28
2.15: Dünyanın yüksekliklere göre renklendirilmiş gösterimi	29
2.16: Sel baskın yüzey modellemesi örneği.....	30
2.17: A ve B noktaları arasındaki görünürlük analiz sonuçları	31
2.18: Ağ örnek gösterimi	32
3.1: Anlamsal Web Mimarisi.....	38
3.2: Bir RDF Çizge Örneği.....	42
4.1: Topolojik İlişkiler.....	54

ŞEKİLLER DİZİNİ (devam)

Sekil

Sayfa

4.2: Topolojik İlişkiler Ontolojisi.....	54
4.3: Örnek bir olgu	56
4.4: Coğrafi Detay Ontolojisinin Ana Şeması	57
5.1: Heterojen uygulamaların birlikte çalışabilirliği kapsamında coğrafi verilerin paylaşımı.....	58
5.2: rootNode Üst-Verisi.....	60
5.3: FuzzyMatchFactor Üst-Verisi.....	61
5.4: maxResults Üst-Verisi	62
5.5: Search Üst-Verisi	62
5.6: Result Üst-Verisi.....	63
5.7: Uygulamanın gösterimi.....	66
5.8: Aramanın yapılması	68
5.9: Bilgi paylaşımı	74

KISALTMALAR DİZİNİ

Kısaltmalar

ADL	Alexandria Digital Library
CBS	Coğrafi Bilgi Sistemleri
DAML+OIL	Defense Advanced Research Projects Agency (DARPA) Agent Markup Language + Ontology Inference Layer
GML	Geographic Markup Language
HTML	Hyper-Text Markup Language
HTTP	Hyper-Text Transfer Protocol
MDNS	Multicast Domain Name Service
MIP	Multilateral Interoperability Programme
OGC	Open Geospatial Consortium
OWL	Web Ontology Language
RDF	Resource Description Framework
RDFS	RDF Schema
SDSS	Spatial Decision Support System
SOAP	Simple Object Access Protocol
URI	Unified Resource Identifier
XML	eXtensible Markup Language
VTYS	Veri Tabanı Yönetim Sistemi
W3C	World Wide Web Consortium
WGS84	World Geodetic System 1984
WSDL	Web Service Definition Language

TEKNİK TERİMLER LİSTESİ

<i>agent</i>	etmen
<i>attribute</i>	öznitelik
<i>cardinality</i>	nitelik çokluğu
<i>class</i>	sınıf
<i>client</i>	istemci
<i>consumer</i>	kullanıcı
<i>constant</i>	sabit
<i>domain</i>	etki alanı
<i>entity</i>	varlık
<i>discovery</i>	keşfetmek
<i>feature</i>	detay
<i>fuzzy</i>	belirsiz/bulanık
<i>geospatial</i>	mekansal, yere ait
<i>graph</i>	çizge
<i>individual</i>	birey
<i>inference</i>	çıkarsama
<i>information</i>	bilgi
<i>instance</i>	olgu/örnek
<i>nested</i>	iç içe
<i>predicate</i>	önerme
<i>producer</i>	üretici
<i>property</i>	nitelik
<i>query</i>	sorgu
<i>raster</i>	hücrese
<i>range</i>	sınır
<i>retrieval</i>	çekim
<i>search</i>	arama
<i>semantic</i>	anlamsal
<i>statement</i>	durum
<i>structured</i>	yapısal
<i>syntactical</i>	sözdizimsel
<i>syntax</i>	sözdizim
<i>topology</i>	topoloji
<i>wrapper</i>	paketleyici

1. GİRİŞ

İnternet, çağımızın yegane en geniş bilgi kaynağı olarak kabul edilmektedir. Buna rağmen, İnternet'in tam elverişli olarak kullanıldığı pek söylenemez. Çünkü, İnternet üzerinde bilgilerin çoğu HTML gibi bilgisayarların okuyabileceği sözdizimsel (*syntactical*) programlama dilleri kullanılarak ifade edilmektedir. Bu gibi programlama dilleri, sadece insanların bilgiyi okuyabilmelerini sağlamaktadırlar. Dolayısıyla, sınırlı imkân ve kabiliyetleri vardır. İnternet denilen bu büyük bilgi kaynağının imkânlarını kısıtlamamak için, bilgisayarların da bilgiyi sadece okumalarını değil, aynı zamanda anlayabilmelerini sağlamak gerekmektedir. Bilgisayarların bilgiyi anlamalarını sağlayabilmek için ise, depolanan bilginin anlamını tanıtabilen anlamsal (*semantic*) programlama dillerine ihtiyaç duyulmaktadır. Ancak, bundan sonra etmenlerin (*agents*-insan veya bilgisayar), bu bilgileri çeşitli şekillerde kullanabilmelerine imkân sağlanmış olacaktır.

Diğer taraftan, bilgisayar donanım ve yazılımında son yıllarda görülen gelişmeler, bilgisayarların diğer bilimsel dallarla olan ilişkilerini de arttırmıştır. Coğrafi Bilgi Sistemleri (CBS) de bu gelişmeler sonunda doğmuş bir bilim dalıdır. Konumla ilgili temel ilişkisi nedeniyle Jeodezi ve Fotogrametri Mühendisliğini de ilgilendirmektedir.

Coğrafi Bilgi Sistemleri, bilgi sistemlerinin grafik bilgiye, çoğunlukla da coğrafi yani mekânsal-konumsal bilgiye dayalı türüdür. Günümüzde oldukça fazla miktarda CBS uygulamaları geliştiren ve pazara sunan ticari şirketler ortaya çıkmaya başlamıştır. Bu uygulamalar ise, kendisine ait formatta veya büyük bir çoğunlukla kamu kuruluşlarının ürettiği formatta coğrafi bilgileri kullanılmaktadırlar. İşte tam bu noktada, CBS'nin karşılaştığı sorun ortaya çıkmaktadır: “Birlikte Çalışılabilirlik”. Üretimi oldukça pahalı olan coğrafi verilerin, normal olarak satın alınması da oldukça maliyetli bir hale gelmektedir. Daha

açıkça ifade etmek gerekirse, coğrafi veri maliyeti, onu kullanan CBS uygulama yazılımlarının maliyetinden kat kat fazla bir hale gelmiştir. Bu sorunun farkına varılmasıyla, yerinde alınan bir karar sonucu, gönüllülük esasıyla Open Geospatial Consortium (OGC) kurulmuştur. Bu konsorsyum, temel sorun olarak coğrafi verilerin birlikte çalışabilirliğini ele almış ve bu konuda çeşitli tanımlamalar yaparak çözüm önerilerinde bulunmaya devam etmektedir. Bu önerilerden birisi de, bir XML uzantısı olan coğrafi işaretleme dili GML (Geographic Markup Language)'dir.

Bunun yanı sıra, günümüzde çok büyük miktarda coğrafi veriler, İnternet üzerinde depolanmakta ve kullanıcılara sunulmaktadır. İnternet üzerinden veri sağlayan şirket, kurum ve kuruluşlar (Ordnance Survey-İngiliz Harita Dairesi vb.), bu konudaki potansiyeli kavramış ve kendi geliştirdikleri coğrafi verilerini GML formatında İnternet üzerinden sunmaya başlamıştır. Ayrıca, günümüzde İnternet üzerinde hizmet veren ADL (Alexandria Digital Library) gibi coğrafi sayısal kütüphaneler de kurulmaktadır. Bütün bu gelişmelere rağmen, veriyi web üzerinde daha akıllı ve otomatik olarak yönetme ve çekme işlevlerini sağlamak için, coğrafi verinin anlamsal olarak zenginleştirilmesi gereksinimi halen mevcuttur.

Şubat 2001 tarihinde, W3C (World Wide Web Consortium) tarafından, web üzerinde verinin anlamsal gösterimini sağlamak amacıyla “Anlamsal Web (Semantic Web)” kavramı ortaya atılmıştır. World Wide Web’in kurucusu Tim Berners Lee, Anlamsal Web’in şimdiki web teknolojisinin bir uzantısı olduğunu, ilave olarak bilginin anlamının daha iyi bir şekilde tanımlandığını, böylelikle bilgisayar ve kullanıcıların daha iyi bir şekilde birlikte çalışabilirliklerinin sağlandığını ifade etmiştir. Anlamsal web’in çalışabilmesi için, bilgisayarlar yapısal (*structured*) bilgiye ve çıkarsama kurallarına (*inference rules*) ihtiyaç duyarlar. İşte bunu başarmanın yolu da, Ontoloji kullanmaktan geçer.

Ontoloji, kısaca bir kavramsallaştırma tanımlanmasıdır, “yapısal bir sözlük” ve “anlamsallık” sağlar. Bunların yardımıyla, bilgisayarın veriyi anlamasını sağlamak için web kaynakları işaretlenebilir.

Tez çalışmasının ikinci bölümünde, CBS'in ve coğrafi verinin kapsamlı tanımı yapılmış, bileşenleri, işlevleri ve modellenmesi anlatılmıştır. Ayrıca, coğrafi veri tiplerinden ve bu verilerin analiz şekilleri ayrıntılarıyla bahsedilmiştir.

Üçüncü bölümde, günümüzde bilgi sistemlerinin geleceği olarak kabul edilen Anlamsal Web ve onun temelindeki ontolojilerden bahsedilmiştir. Ayrıca, ontoloji özellikleri ile geliştirmek için gerekli olan RDF ve RDFS dilleri hakkında ayrıntılı bilgi verilmiştir.

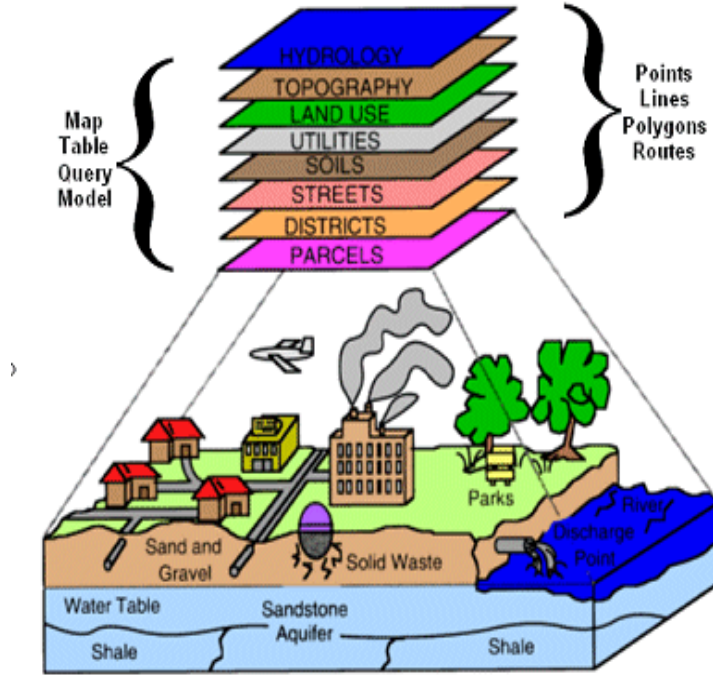
Dördüncü bölümde, günümüzde ismi yeni yeni duyulmaya başlanan ve anlamsal web ile CBS'nin bir bileşimi olan Anlamsal CBS anlatılmıştır. Bu kapsamda, GML ve coğrafi ontolojilerden bahsedilmiş, coğrafi ontolojilerin nasıl ve ne amaçla geliştirildiği konularına ayrıntılarıyla değinilmiştir. Ontoloji yaratma araç yazılımları Protégé ve SWOOP anlatılmıştır. Her iki araç yazılımının farklılıkları ve birbirine olan üstünlükleri ifade edilmeye çalışılmıştır. Kısaca özetlemek gerekirse, bu çalışmalarda temel amacın anlamsal coğrafi veriler elde edilmesi olduğu ve böylelikle verilerin anlamsal olarak sorgulanabilecek bir hale getirildiği, bunun sonucunda bir özelleştirilmiş çıkarsama motoru tasarımı yapılabileceğinden bahsedilmiştir.

Beşinci bölümde, coğrafi verilerin heterojen sistemler arasında anlamsal web teknolojisi kullanarak değişimini sağlamak maksadıyla, proje kapsamında gerçekleştirilen uygulama, ayrıntılarıyla anlatılmıştır.

Son olarak altıncı bölümde ise, tez çalışmasında elde edilen sonuçlar değerlendirilmiş ve anlamsal coğrafi bilgi sistemlerinin gelecekteki yeri tartışılmıştır.

2. COĞRAFI BİLGİ SİSTEMİ VE COĞRAFI VERİ

Coğrafi Bilgi Sistemi (CBS, Geographic Information System: GIS), karmaşık planlama ve yönetim sorunlarının çözülebilmesi için tasarlanan; coğrafi konumu belirli verilerin toplanması, depolanması, yönetimi, işlenmesi, analiz edilmesi, modellenmesi ve görsel olarak sunulması işlemlerini kapsayan donanım, yazılım, personel ve yöntemler sistemidir. Başka bir deyişle Coğrafi Bilgi Sistemi, konumsal ve konumsal olmayan verilerin, yönetim, personel, yazılım, donanım yardımıyla; veri değişim standartlarına uygun şekilde bir araya getirilerek depolanması, analiz edilmesi, sorgulanması, kullanıcıların isteklerine cevap verecek biçimde tasarlanması ve hizmete sunulması ile oluşan teknolojik bir sistemdir.

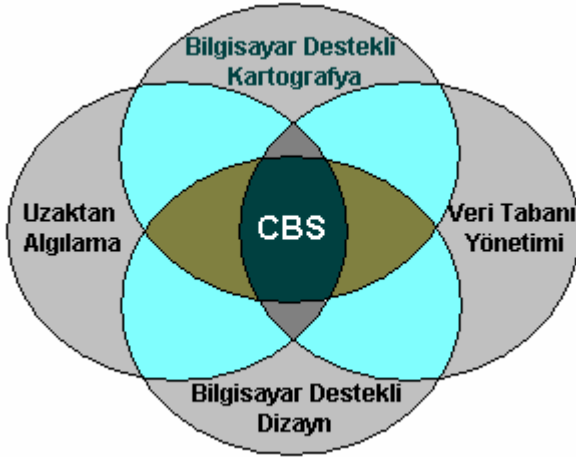


Şekil 2.1: CBS'nin gerçek dünya ile ilişkilendirilmesi
(<http://www.cnr.vt.edu>, 2006)

CBS’de katmanların oluşturulabilmesi için coğrafi verilerin yeryüzü üzerinden çeşitli yollarla toplanarak sayısal formata çevrilmesi ve kullanıcıya sunulması işlemi Şekil 2.1’de ifade edilmektedir.

Coğrafi veri, belirli bir amaç için kullanılmak üzere elde edilmiş dünya üzerinde varolan bir mekanın grafiksel ve grafiksel olmayan bileşenlerinin tümüdür. Grafiksel veri bu mekanın görsel olan ve harita üzerinde gösterilen bileşenidir. Grafiksel olmayan veri ise, bu mekana ilişkin görsel olmayan fakat niteliği olan bileşendir. Örneğin, bir fiziki harita üzerinde mavi çizgiler ile gösterilen nehirler grafiksel veri, bu nehirlerin uzunluğu, derinliği, debisi ise grafiksel olmayan veridir.

CBS, Şekil 2.2’de görüldüğü gibi Uzaktan Algılama, Bilgisayar Destekli Tasarım, Veri Tabanı Yönetim Sistemi ve Bilgisayar Destekli Kartoğrafya gibi teknolojileri bütünleşik olarak kullanan bir sistemdir.



Şekil 2.2: CBS ile diğer Bilgi Sistemleri arasındaki ilişki (Tecim, 2001).

İyi tasarlanmış bir CBS aşağıda sıralanan özelliklere sahip olmalıdır (Yomralıoğlu, 2000):

- Büyük boyuttaki verilerle hızlı ve kolay bir şekilde işlem yapabilmelidir.
- Farklı veri katmanlarını birbirleri ile ilişkilendirebilmelidir.
- Verileri konumsal ve konumsal olmayan özellikleri ile analiz edebilmelidir.
- Belirli bir özellik veya öznitelik için sorgulama ve arama yetenekleri olmalıdır.
- Veriyi hızlı ve düşük maliyetli bir şekilde güncelleyebilmelidir.
- Veriyi modelleme ve alternatifleri değerlendirme yeteneğine sahip olmalıdır.
- İhtiyaçları karşılayacak her türlü harita, rapor formatında çıktılar elde edebilmelidir.

2.1. Coğrafi Bilgi Sistemi Bileşenleri

CBS, temel olarak Şekil 2.3’de gösterilen beş ana bileşenden oluşmaktadır (Tecim, 2001):

- Veri
- Donanım
- Yazılım
- İnsanlar
- Metotlar

CBS’nin en önemli bileşeni **veridir**. Grafik yapıdaki coğrafi veriler ile tanımlayıcı nitelikteki öznitelik veya tablo verileri ilgili kaynaklardan toplanabileceği gibi, piyasada bulunan hazır haldeki veriler de satın alınabilir. CBS konumsal veriyi diğer veri kaynaklarıyla

birleştirebilir. Böylece birçok kurum ve kuruluşa ait veriler organize edilerek konumsal veriler bütünleştirilmektedir.

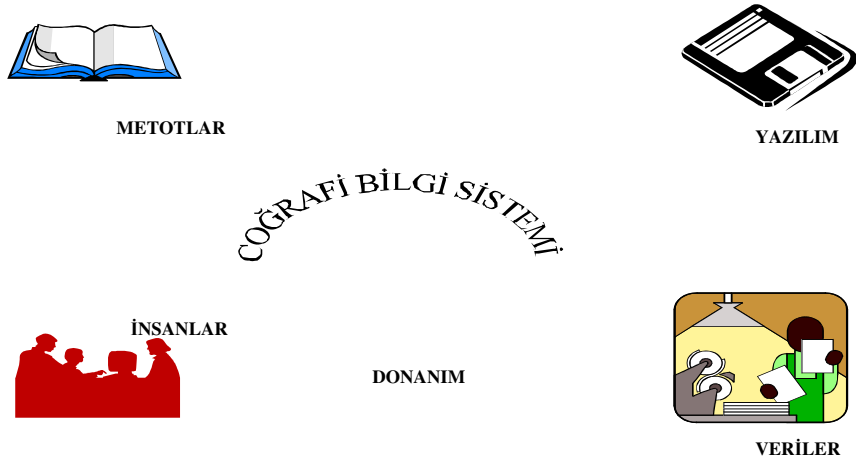
Veri, uzmanlarca CBS için temel öge olarak kabul edilirken, elde edilmesi en zor bileşen olarak da görülmektedir. Veri kaynaklarının dağınıklığı, çokluğu ve farklı yapılarda olmaları, bu verilerin toplanması için büyük zaman ve maliyet gerektirmektedir. Nitekim, CBS'ye yönelik kurulması tasarlanan bir sistem için harcanacak zaman ve maliyetin yaklaşık %50'den fazlası veri toplamak için gerekmektedir (Yomralıoğlu, 2000).

CBS'nin işlemlerini mümkün kılan bilgisayar ve buna bağlı yan ürünlerin bütünü **donanım** olarak adlandırılır. Bütün sistem içerisinde en önemli araç olarak gözüken bilgisayar yanında yazıcı, tarayıcı, sayısallaştırma masası gibi yan donanımlara da ihtiyaç vardır.

Yazılım, diğer bir deyişle bilgisayarda çalışan program, coğrafi bilgi girişi yapmak, depolamak, sorgulamak, analiz etmek ve görüntülemek gibi ihtiyaç ve fonksiyonları kullanıcıya sağlamak üzere geliştirilen algoritmalarlardır. Yazılımların pek çoğunun ticari amaçlı firmalarca geliştirilip üretilmesi yanında üniversite ve benzeri araştırma kurumlarınca da eğitim ve araştırmaya yönelik geliştirilmiş yazılımlar da mevcuttur. Dünyadaki CBS pazarının önemli bir kısmı yazılım geliştiren firmaların elindedir. Bu bakımdan günümüzde CBS bu tür yazılımlarla neredeyse özdeşleşmiş durumdadır.

CBS teknolojisi **insanlar** olmadan sınırlı bir yapıda olurdu. Çünkü insanlar, gerçek dünyadaki problemleri uygulamak üzere gerekli sistemleri yönetir ve gelişme planları hazırlarlar. CBS kullanıcıları, sistemleri tasarlayan ve koruyan uzman teknisyenlerden, günlük işlerindeki performanslarını artırmak için bu sistemleri kullanan kişilerden oluşan geniş bir kitledir. Dolayısıyla, coğrafi bilgi

sistemlerinde insanların istekleri ve yine insanların bu istekleri karşılamaları gibi bir süreç yaşanır. CBS'nin gelişmesi mutlak suretle insanların yani kullanıcıların ona sahip çıkmalarına ve konuma bağlı her türlü analiz için CBS'yi kullanabilme yeteneklerini artırmaya ve değişik disiplinlere CBS'nin avantajlarını tanıtmakla mümkün olabilecektir.



Şekil 2.3: Coğrafi Bilgi Sistemi Temel Bileşenleri.

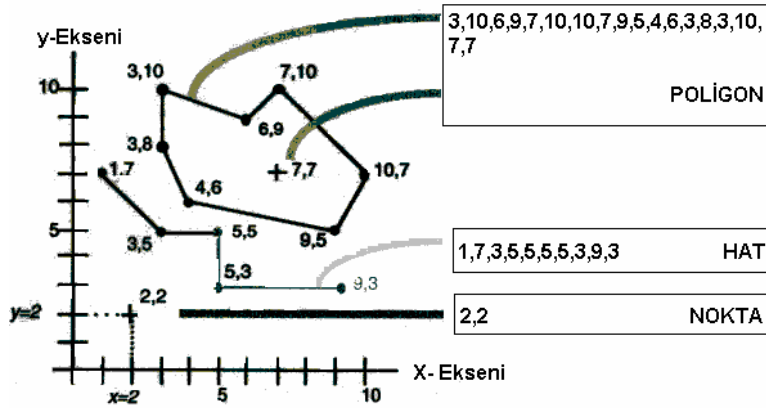
Başarılı bir CBS, çok iyi tasarlanmış plan ve iş kurallarına göre işler. Bu tür işlevler her kuruma özgü model ve uygulamalar şeklindedir. CBS'nin kurumlar içerisindeki birimler veya kurumlar arasındaki konumsal bilgi akışının verimli bir şekilde sağlanabilmesi için gerekli kuralların yani **metotların** geliştirilerek uygulanıyor olması gerekir. Konuma dayalı verilerin elde edilerek kullanıcı talebine göre üretilmesi ve sunulması mutlaka belirli standartlar yani kurallar çerçevesinde gerçekleşir. Genellikle standartların tespiti şeklinde olan bu uygulamalar bir bakıma kurumun yapısal organizasyonu ile doğrudan ilgilidir. Bu amaçla yasal düzenlemelere gidilerek gerekli yönetmelikler hazırlanarak ilkeler tespit edilir.

2.2. Grafiksel Veri ve Modellenmesi

Konumsal bilgiler, enlem-boylam şeklindeki coğrafi koordinat gibi kesin değerleri veya adres, bölge ismi, yol ismi gibi tanımlanan referans bilgileri içerirler. Bu coğrafi referanslar nesnelerin konumlandırılmasına, yani koordinatı bilinen bir pozisyona yerleştirilmelerine imkan sağlar. Böylece, ticari bölgeler, araziler, orman alanları, yeryüzü kabuk hareketleri ve yüzey şekillerinin analizleri konuma bağlı olarak belirlenir. Coğrafi referans konumu belirlerken, konum verisi yani koordinat bilgisi seçilecek veri modeline bağlı olarak ifade edilir. Bu ifade şekli CBS’de iki farklı konumsal veri modeli biçimindedir. Bunlar **vektörel** ve **hücrel (raster)** veri modelleridir.

Şekil 2.4’de görüldüğü gibi vektörel veri modelinde, nokta, çizgi ve poligonlar (x,y) koordinat değerleriyle kodlanarak depolanırlar. Nokta özelliği gösteren bir elektrik direği tek bir (x,y) koordinatı ile tanımlanırken, çizgi özelliği gösteren bir yol veya akarsu şeklindeki coğrafi detay birbirini izleyen bir dizi (x1,y1) (x2,y2) ... (xn,yn) koordinat serisi şeklinde saklanır. Poligon özelliğine sahip coğrafi veriler, örneğin imar adası, bina, orman alanı, parsel veya göl, kapalı şekiller olarak, başlangıç ve bitişinde aynı koordinat olan (x1,y1) (x2,y2) ... (xn,yn) (x1,y1) dizi koordinatlar ile depolanır.

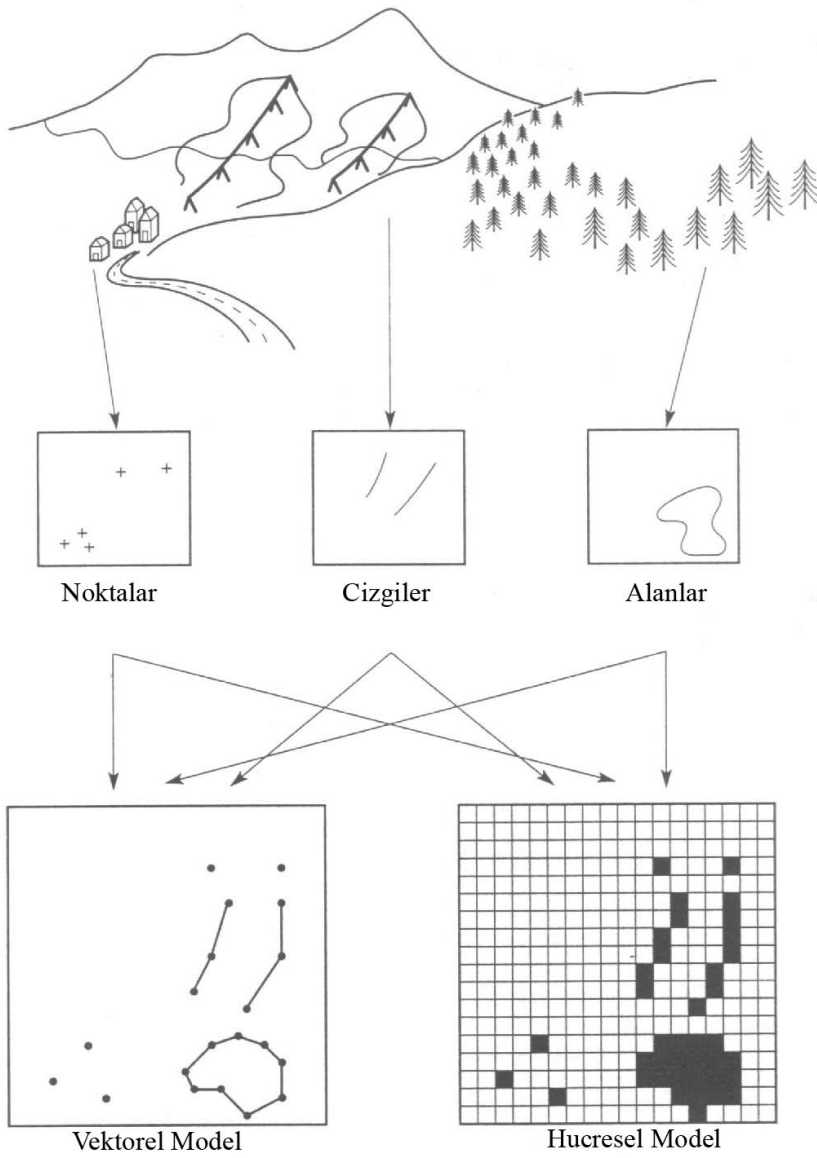
Vektörel model coğrafi varlıkların kesin konumlarını tanımlamada son derece yararlı bir modeldir. Ancak, süreklilik özelliği gösteren coğrafi varlıkların, örneğin toprak yapısı, bitki örtüsü, jeolojik yapı ve yüzey özelliklerindeki değişimlerin ifadesinde daha az kullanışlı bir model olarak bilinir.



Şekil 2.4: Uzaysal verilerin vektörel olarak gösterimi ve x, y koordinatlarına dönüştürülmesi (ESRI, 1997).

Şekil 2.5, vektörel ve hüresel veri modellerinin gerçek dünyadan alınan ve CBS'ye nokta, çizgi, alan olarak aktarılan coğrafi nesneleri nasıl modellediğini göstermektedir.

Hüresel ya da diğer bir deyişle raster veri modeli daha çok süreklilik özelliğine sahip coğrafi varlıkların ifadesinde kullanılmaktadır. Raster görüntü, birbirine komşu grid yapıdaki aynı boyutlu hücrelerin bir araya gelmesiyle oluşur. Hücrelerin her biri piksel olarak da bilinir. Fotoğraf görüntüsü özelliğine sahip raster modeller, genellikle fotoğraf ya da haritaların taranması ile elde edilirler. Vektör ve raster veri modellerinden biri genelde CBS uygulama biçimine göre tercih edilerek kullanılır. Ancak günümüzde her iki model aynı anda da kullanılabilir.

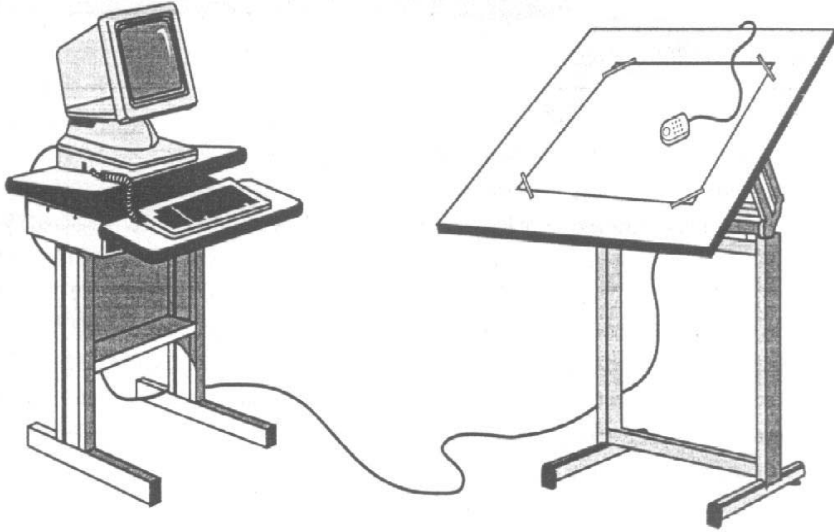


Şekil 2.5: Grafiksel veri modelleri (Clarke, 2001).

2.3. Coğrafi Bilgi Sistemi İşlevleri

2.3.1. Veri Toplama

Coğrafi veriler toplanarak, CBS’de kullanılmadan önce mutlaka sayısal formata dönüştürülmelidir. Verilerin kâğıt ya da harita ortamından bilgisayar ortamına dönüştürülmesi işlemine sayısallaştırma denir. Modern CBS teknolojisinde bu tür işlemler büyük boyutlu projelerde tarama tekniği kullanılarak otomatik araçlarla gerçekleşir. Küçük boyutlu projelerde daha çok masa tipi sayısallaştırıcılar kullanılarak elle sayısallaştırma yapılabilir. Bugün birçok coğrafi veri CBS’ye uyumlu formatta hazır halde piyasada mevcuttur. Bunlar üretici firmalardan sağlanarak doğrudan kurulacak sisteme aktarılabilir. Şekil 2.6 ve 2.7 CBS için farklı veri toplama yollarına örnek vermektedir.



Şekil 2.6: Bilgisayara bağlanmış bir sayısallaştırma masası (Clarke, 2001).



Şekil 2.7: Uydu fotoğrafları otomatik veri toplama yöntemlerinden biridir (<http://www.lapanse.com>, 2006).

2.3.2. Veri Depolama

Küçük boyutlu CBS projelerinde coğrafi bilgilerin sınırlı boyuttaki basit dosyalarda saklanması mümkündür. Ancak, veri hacimlerinin geniş ve kapsamlı olması, bunun yanında birden çok veri gruplarının kullanılması durumunda Veritabanı Yönetim Sistemleri (VTYS) verilerin saklanması, organize edilmesi ve yönetilmesine yardımcı olur. VTYS bir bilgisayar yazılımı olup veritabanlarını yönetir veya birleştirir. Bir çok yapıda tasarlanmış veritabanı yönetim sistemi vardır, ancak CBS için en kullanışlı ilişkisel veritabanı sistemidir. Bu sistem tasarımında veriler tablo bilgilerinin elde edilişindeki düşünce yapısına uygun olarak

bilgisayar belleğinde saklanır. Farklı bilgiler içeren tabloların birbiriyle ilişkilendirilmesinde bu tablolardaki ortak sütunlar kullanılır. Bu yaklaşım basit fakat esnek bir tasarım olup, geniş çapta CBS uygulamalarında kullanılmaktadır.

2.3.3. Veri İşleme ve Dönüşüm

Bazı durumlarda özel CBS projeleri için veri çeşitlerinin birbirine dönüşümü istenebilir. Verilerin sisteme uyumlu olması bunu gerektirebilir. Örneğin, konumsal bilgiler farklı ölçeklerde mevcut olabilir (yol verileri 1/100.000, nüfus dağılım verileri 1/10.000, bina verileri 1/1.000 gibi). Tüm bu bilgiler birleştirilmeden önce aynı ölçeğe dönüştürülmelidir. Bu dönüşüm görüntü amacıyla geçici olabileceği gibi bir analiz işlemi için sürekli ve kalıcı da olabilir. CBS, gerek bilgisayar ortamında nesne üzerine imlecin tıklanması ile basit sorgulama kapasitesine, gerekse çok yönlü konumsal analiz araçlarıyla yönetici ve araştırmacılara istenen süreçte bilgi sunar. CBS teknolojisi, artık coğrafi verileri istatistiksel grafikler ve “eğer olur ise..” şeklindeki mantık sorgulamaları ve senaryolar şeklinde irdeleme aşamasına gelmiştir. CBS teknolojisi kullanarak konumsal verilerin sorgulanması ve analizinde, yazılımlar sayesinde, birçok veri her türlü geometrik ve mantıksal işleme tabi tutulabilir.

2.3.4. Veri Analizi

Karar verme aşamasında, toplanmış ve sisteme aktarılmış coğrafi verilerin analiz edilmesi ile çeşitli değerlendirmeler yapılabilir. Coğrafi analizler en basit şekliyle; imleç ile seçilen nesnelerin özelliklerinin sorgulanmasından, uzunluk, alan, hacim ölçümlerine, katmanların çakıştırılarak yeni katmanlar elde edilmesine, arazinin üç boyutlu görüntülenmesine kadar uzanabilir. Analizler neticesinde elde edilen somut sonuçlar rapor haline getirilebilir, haritalar oluşturularak görsel

olarak sunulabilir. Bu sayede CBS konusunda uzman olmayan kişilerin de bu sonuçlardan faydalanmaları sağlanabilir.

Genellikle büyük projelerde karar verme yetkisine sahip kişiler, CBS uzmanları değil, ilgili konunun uzmanlarıdır. Bu nedenle analizlerden elde edilen tekrarlanabilir ölçüm ve sonuçlar karar verme aşamasında bu kişiler için değerlidir.

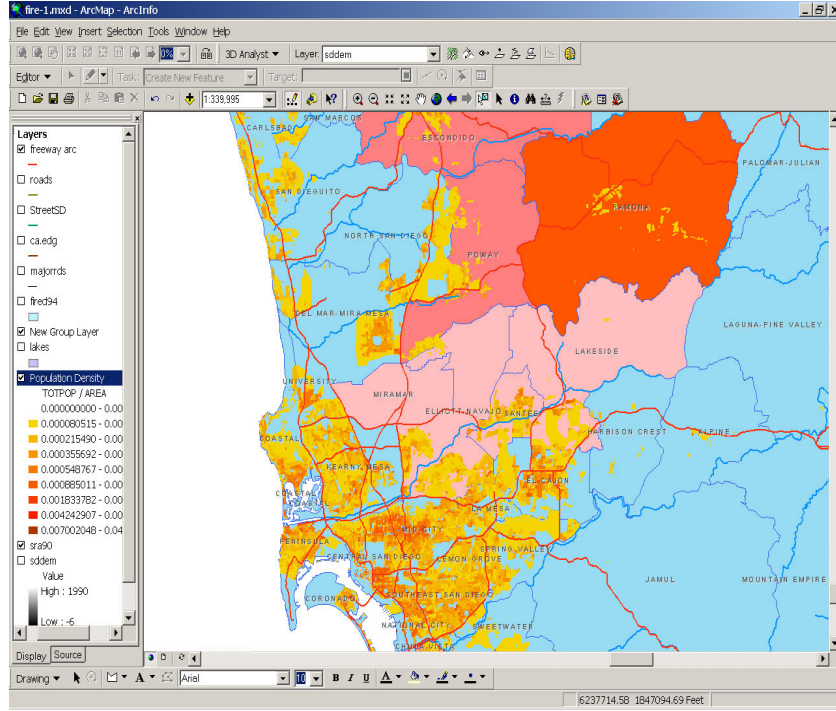
2.3.5. Veri Sunumu (Gösterim)

Görsel işlemler, CBS için önemli bir işlevdir. Birçok coğrafi işlemin sonunda yapılanlar, harita veya grafik gösterimlerle görsel hale getirilir. Haritalar, coğrafi bilgiler ile kullanıcı arasındaki en iyi iletişimi sağlayan araçlardır. Kartoğrafların uzun yıllardır harita üretmesine karşın, CBS kartoğrafya biliminin hızlı gelişmesine de katkıda bulunan yeni ve daha etkili araçları sunmaktadır. Haritalar, yazılı raporlarla, üç boyutlu gösterimlerle, fotoğraf görüntüleri, çoklu-ortamlı (multimedia) ve diğer çıktı çeşitleriyle birleştirilebilmektedir. Şekil 2.8’de bir CBS uygulaması ile dünyanın bir kesiminin yükseklik ve siyasi ülke sınırları ile gösterimi sunulmaktadır.

2.4. Coğrafi Veri ve Analizler

2.4.1. Konum Belirleme

Varolan her nesne dünya üzerinde belirli bir konuma sahiptir. Bir nesnenin harita üzerinde gösterilebilmesi için konumunun koordinat sistemi üzerinde ifade edilmesi gerekir. Bu nesnenin konumunun kesinliği koordinat sistemine bağlıdır.



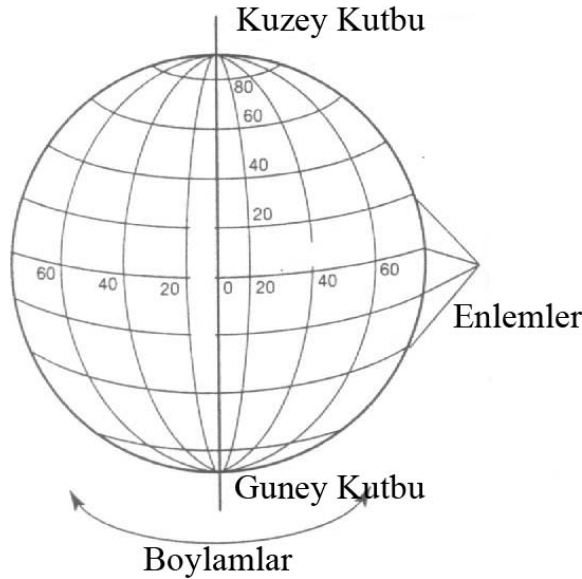
Şekil 2.8: CBS’de verilerin sunumu (<http://map.sdsu.edu>, 2006)

Farklı veri kaynaklarından elde edilen nesnelerin konumları farklı koordinat sistemleri üzerinde tanımlanmış olabilir, bu verileri veri işleme ve dönüşüm işlemlerinden geçirdikten sonra ortak bir koordinat sistemi üzerinde göstermek CBS’nin görevlerindendir.

Konumun belirlenmesi CBS analizlerinin yapılabilmesi için ilk adımdır. Önceden tanımlanmış ve sıklıkla kullanılan koordinat sistemleri olduğu gibi CBS kullanıcısı gerekli kuralları belirleyerek kullanıcı tanımlı bir koordinat sistemi de oluşturabilir. CBS’de en sık kullanılan koordinat sistemi Coğrafi Grid Sistemi’dir.

Coğrafi Grid Sistemi’nde bir yerin konumu dünya üzerinde ekvatora ve baş meridyene olan uzaklığı ile belirlenir. Coğrafi kuzey (kuzey kutbu) dünyanın dönüş ekseninin kuzey ucu, coğrafi güney

(güney kutbu) ise bu eksenin güney ucu olarak tanımlanır. Ekvator dünya yüzeyinde coğrafi kuzey ve coğrafi güneyin tam ortasından geçen hayali çemberdir. Paralel (enlem) dünya yüzeyinde ekvatora paralel olan herhangi bir çemberdir. Meridyen (boylam) coğrafi kuzey ve coğrafi güneyden geçen ve paralelleri dik olarak kesen herhangi bir çemberdir. Baş meridyen Greenwich'teki İngiliz Kraliyet Gözlemevi'nden geçen meridyen olarak kabul edilir. Şekil 2.9'da Coğrafi Grid Sistemi'nin temel bileşen ve kavramları gösterilmiştir.



Şekil 2.9: Coğrafi Grid Sistemi (Clarke, 2001)

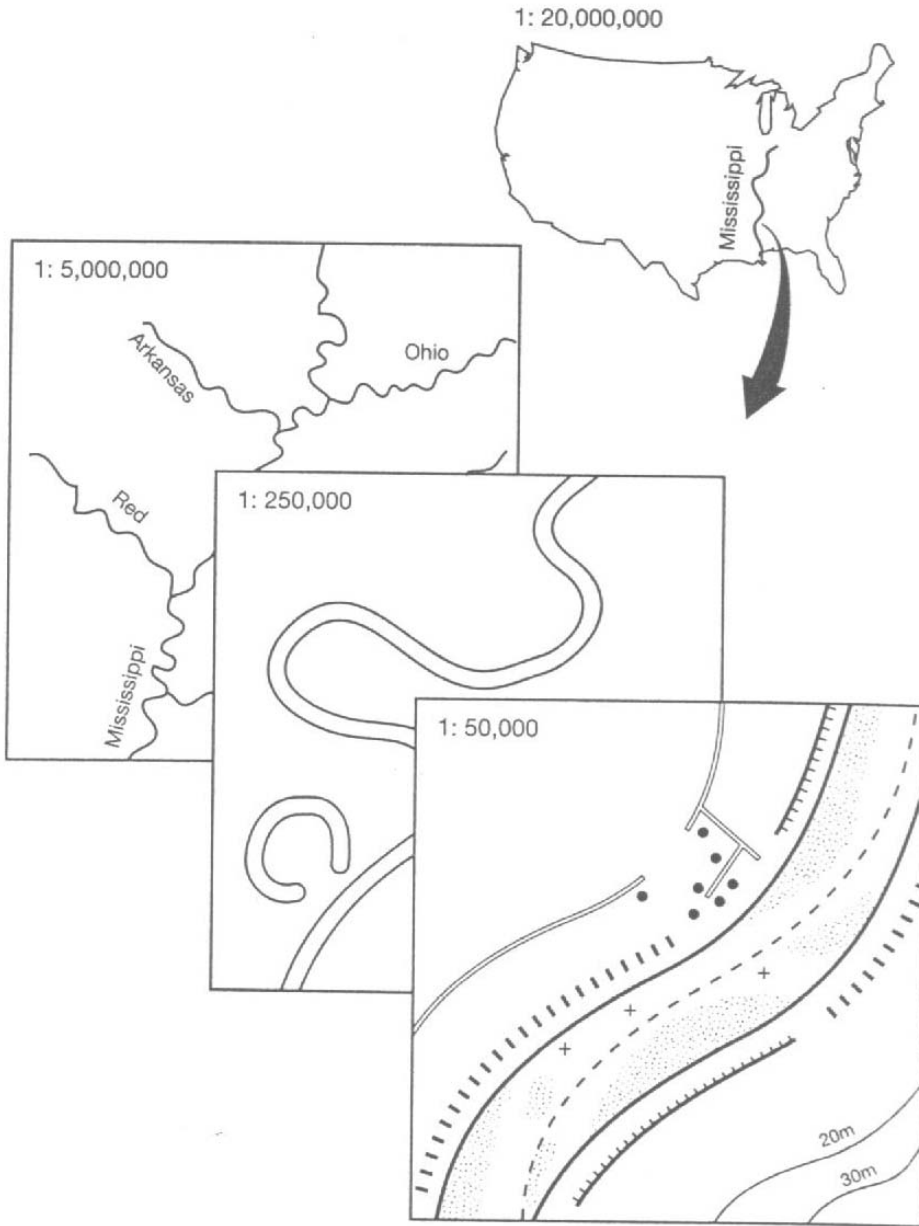
Dünya yüzeyindeki herhangi bir noktanın coğrafi koordinatları Coğrafi Grid Sistemi'nde bu noktanın dünyanın merkezi ile ekvator ve baş meridyen arasındaki açısal uzaklığına bağlı olarak belirlenir. Ekvator 0 derece enlem, baş meridyen 0 derece boylam olarak kabul edilir.

Harita ölçeği de koordinat sistemini tanımlayan diğer bir parametredir. Ölçek, haritadaki birim uzunluğun yeryüzündeki gerçek

uzunluęuna oranı olarak tanımlanır. Haritalarda ölçek matematiksel olarak (1/5000 gibi) veya grafiksel olarak gösterilebilir. Büyük ölçekli haritalar küçük alanları detaylı olarak gösterirken küçük ölçekli haritalar büyük alanları az detaylı olarak gösterirler.

Şekil 2.10'da aynı bölgenin farklı ölçeklerde verilen haritalarından gösterilen detayların harita ölçeęi ile nasıl bir ilişkisi olduęu açıkça görölmektedir.

Harita Projeksiyonu: Koordinat sistemini tanımlayan en önemli parametrelerden birisi ise haritanın projeksiyonudur. Projeksiyon küre olarak kabul edilebilecek olan üç boyutlu dünya yüzeyinin iki boyutlu haritaya aktarılması için gerekli olan matematiksel tanımlamalardır. Bu amaçla birçok projeksiyon tanımlanmıştır. Ancak, tanımlanan herbir projeksiyon yeryüzünün şekil, alan, uzaklık, yön gibi farklı bir özellięinin yanlış tanımlanmasına neden olur. Üç boyutlu yeryüzünü iki boyutlu haritaya indirgerken bu özelliklerden bir veya birkaçının bozulması kaçınılmazdır. Bu sebepten dolayı istenen haritanın kullanım amacına göre korunması gereken özellikleri koruyan ve hedeflenen bölgeyi en iyi şekilde ifade edebilen projeksiyon seçilmelidir veya eldeki veri bu projeksiyona dönüştürölmelidir. Yapılacak tüm konumsal analizler belirlenen koordinat sisteminden etkileneceęi için bu seçimin yapılması için gerekli araştırmalar yapılmalı, uzman görüşü ve önceki çalışmalar dikkate alınmalıdır.



Şekil 2.10: Harita ölçekleri ve gösterilen detaylar (Heywood et al., 2002).

2.4.2. Temel Coğrafi Nesneler

Coğrafi nesneler temel olarak nokta, çizgi (hat) ve alan (poligon) sembollerinden oluşur (Şekil 2.3). Nokta sembolleri, gösterimde alan olarak ifade edilmek için çok küçük olan nesneler için kullanılır. Örneğin, bir enerji iletim hattındaki elektrik direkleri veya bir altyapı haritasındaki logar kapakları nokta ile gösterilebilecek gerçek nesnelerdir. Bir nokta, coğrafi konumunu ve yüksekliğini belirleyen (x, y, z) şeklinde ifade edilebilir.

Çizgi sembolleri gerçekte doğrusal olan enerji iletim hattı, nehir, yol gibi nesneler için kullanılır. Çizgi sembolleri kırılma noktalarının meydana getirdiği birden çok parçadan oluşabilir ancak her bir parça doğrusaldır. Çizgi sembollerinin anlamlı bir şekilde birliktelik oluşturduğu durumlar olabilir. Buna örnek olarak bir şehir merkezindeki caddeler verilebilir. Bu tür çizgilerden oluşan yapılara ağ (network) adı verilir. Bu yapılar üzerinde uygulanabilecek farklı CBS analizleri mevcuttur.

Alan sembolleri gerçekte göl, arsa gibi gösterimde belirli bir alana sahip olan nesneler için kullanılır. Alanlar, çizgilerin kapalı bir bölge oluşturması sonucunda ortaya çıkarlar. Bu alanlar birbirine komşu olabildikleri gibi birbirlerinin içinde de yer alabilirler, örneğin bir göl içerisindeki ada gibi.

2.4.3. Temel Ölçümler

CBS'deki temel ölçümler iki nokta arasındaki uzaklığın ölçülmesi, alan ve hacim ölçülmesi gibi işlemlerdir. İki nokta arasındaki uzaklık en basit şekliyle matematiksel olarak bulunabilir. Ancak, CBS'de amaç gerçek dünya üzerindeki nesneler arasındaki uzaklığı ölçmek olduğuna göre buradaki uzaklık ölçümü dünyanın yuvarlaklığının dikkate alındığı ölçümdür.

Aynı şekilde alan en basit şekliyle matematiksel olarak bulunabilir. Ancak, CBS’de amaç alanı dünyanın yuvarlaklığını dikkate alarak yüzey alanı olarak bulmaktır.

2.4.4. Veri Sorgulama

CBS’nin asıl amacı verilerin gösterilmesi değil, bu verilerin sorgulanabilmesi, raporlanabilmesi ve üzerinde analizler yapılabilmesidir. Veri sorgulama CBS’nin temel fonksiyonu ve belki de en sık kullanılanıdır. Değişik veri kaynaklarından gelen ve aynı bölgeye ait olan veriler ayrı katmanlar üzerinde tutulur. Bu katmanların herbirisi ayrı bir asetat kağıdı gibi de düşünülebilir. Herhangi bir zamanda istenilen katmanın görsel olarak açılıp kapatılması ve dolayısıyla CBS’de görülen bilgilerin değiştirilmesi mümkündür.

Her katmandaki veri, grafiksel olduğu gibi aynı zamanda bazı öznitelik (grafiksel olmayan) bilgilerine de sahip olabilir. Örneğin, Türkiye’deki illerin sınırlarını gösteren bir harita katmanı düşünelim. Bu katmandaki grafiksel veri, yani iller, alan sembolü ile gösterilmektedir. Her bir sembol, yani il görünen nesnenin ardında toplam nüfus, il plaka numarası, ilin adı gibi öznitelik bilgilerine de sahiptir. CBS’deki sorgulama aracı ile seçilen herhangi bir ilin bu öznitelik bilgilerine ulaşmak, bunları raporlamak ve hatta bu bilgilere dayanarak tematik görüntüler yaratmak (örneğin illerin belirli nüfus aralıklarına göre renklendirilerek gösterilmesi gibi) CBS’nin sağladığı yeteneklerdir.

Belirli öznitelik bilgilerine göre nesnelerin seçilmesi yani belirlenen bir kritere göre seçim yapılması filtreleme olarak adlandırılır ve yine CBS’nin sağladığı temel yeteneklerdendir. Filtreleme yeteneği sayesinde istenmeyen nesneler görüntüden kaldırılabilir ve daha anlaşılır ve sade bir görünüm elde edilebilir. Bu işlem aslında varolan veride bir

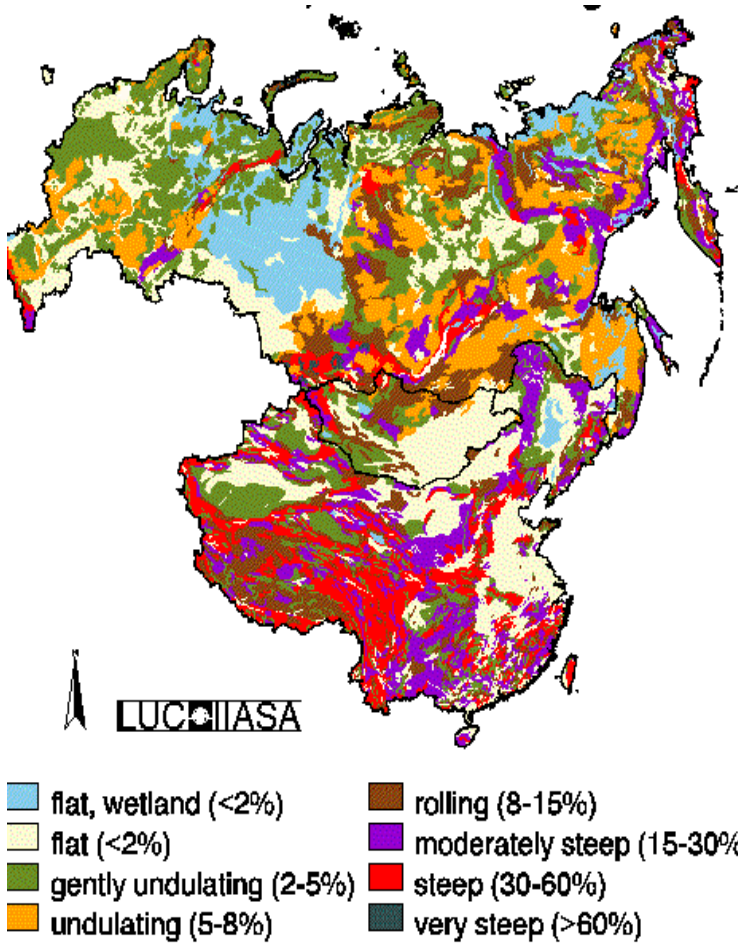
değişiklik yapmamakta sadece görsel olarak yansıtılan resmi değiştirmektedir.

2.4.5. Veri Sınıflandırma

Verilerin öznitelik bilgilerinden bir veya birkaçına göre sınıflandırılıp gruplanması bazı durumlarda verinin daha iyi yorumlanmasını sağlayabilir. Sınıflandırma işlemi için öncelikle verinin hangi öznitelik bilgisine göre kategorilere ayrılacağı belirlenmelidir. Örneğin, Türkiye'deki iller bulundukları bölgelere göre, nüfus yoğunluklarına göre, yüzölçümlerinin büyüklüğüne göre sınıflandırılabilir.

Daha sonra verinin kaç grupta sınıflandırılacağına karar verilmelidir. Kaynaklar, insanların üç ile yedi arasındaki farklı sınıflı rahatlıkla algılayabildiğini, daha fazla sayıda sınıfın algılamayı zorlaştırdığını belirtmektedir (Esengin, 2004). En son olarak da sınıflandırmadaki yöntem belirlenmelidir. Örneğin, illerin nüfuslarına göre sınıflandırılması eşit nüfus aralıklarına göre, kullanıcı tanımlı nüfus aralıklarına göre, hatta matematiksel bir dağılıma göre yapılabilir. Bu, sınıflandırmanın amacına ve elde edilmek istenen sonuca bağlıdır. Şekil 2.11'de dünya ülkeleri farklı renklerle gösterilmiştir.

Sınıflandırma işlemi hem vektörel hem de hücresel veriler üzerinde rahatlıkla uygulanabilir. Tematik haritaların elde edilmesi de aslında bir çeşit sınıflandırmanın sonucudur. Ancak, sınıflandırma sadece görsel sonuçlar elde etmek için değil, eldeki veri kümelerinden yeni veri kümeleri üretmek için de kullanılır. Sınıflandırma işlemi sonucunda tamamen farklı değerlere sahip olan nesneler gruplanarak daha kolay algılanabilir veya işlenebilir hale getirilebilir.



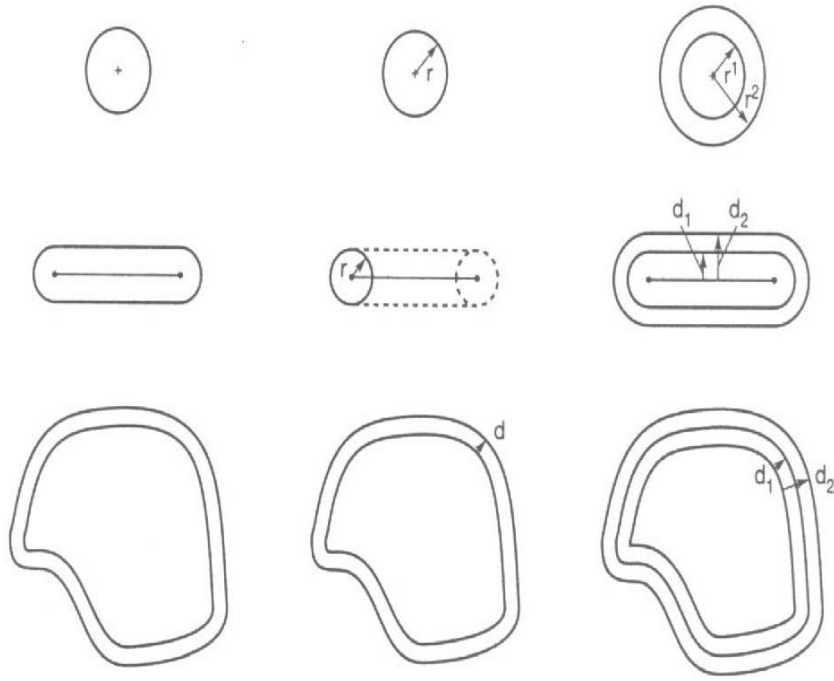
Şekil 2.11: Verilerin sınıflandırılması (<http://www.iiasa.ac.at>, 2006)

2.4.6. Tampon Bölge Oluşturma

Tampon bölge oluşturma genellikle bir nesnenin veya olgunun çevresinde bir alan oluşturmak için kullanılır. Bu nesne veya olgu nokta, çizgi veya alan ile ifade edilmiş olabilir. Tüm bu coğrafi nesnelerin çevresinde tampon bölge oluşturulabilir.

Tampon bölge nesneden belirlenen bir uzaklıktaki tüm bölgeyi kaplayacak şekilde veya nesnenin belirli kısımlarından farklı uzaklıkları

kapsayacak şekilde oluşturulabilir. Bir noktanın tampon bölgesi genellikle bir daire, bir çizginin veya alanınki ise şekle bağlı olan bir alandır. Şekil 2.12’de nokta, çizgi ve alanlar için tampon bölgenin nasıl oluşturulduğu resmedilmektedir.



Şekil 2.12: Nokta, çizgi ve alan sembolleri için tampon bölge oluşturma örnekleri (Heywood et al., 2002)

Tampon bölge oluşturma işlemi belirli kriterler verilerek istenen bölgeleri bulmak için kullanılır. Örneğin, yollara, nehirlere, göllere ve denizlere olan en çok ve en az uzaklık verilerek istenen bölgeler bulunabilir (Tecim, 1997; Komesli, 1999; Cowen et al., 1988). Ayrıca, bir su kaynağı (rezervuar) çevresinde koruma alanları oluşturulabilir, bir yol çevresinde araçlardan çıkan gazla insan hayatını etkileyecek şekilde

kirletilen bölge belirlenebilir veya bir fabrikadan çıkan zehirli gazların dağılırken izlediği ve etki altında bıraktığı alan oluşturulabilir.

2.4.7. Katmanların Çakıştırılması

İki farklı veri kaynağından gelen verileri katman çakıştırma ile bütünleştirebilme yeteneği, CBS'nin kilit fonksiyonudur. Katmanların çakıştırılması haritacılıkta aynı bölgenin farklı özelliklerini birleştirmek için eskiden beri kullanılmakta olan bir yöntemdir. CBS bu işlemi otomatikleştirmekte ve kolaylık, hız, esneklik gibi açılardan iyileştirme sağlamaktadır.

Katmanların çakıştırılması varolan veri kümelerinden yeni bir veri kümesi oluşturmak amacı ile kullanılabilir. Örneğin, ana yollara en fazla 200 metre uzaklıkta bulunan otelleri bulmak istediğimizde elimizde bulunan oteller veri kümesi ile ana yollar veri seti katmanlarını çakıştırma yoluna gidebiliriz. Bu amaçla ana yollar veri kümesinde 200 metre ile bir tampon bölge oluşturarak bu yeni temayı noktasal oteller ile birleştirdiğimizde karşımıza çıkan yeni veri kümesinde otellerin bu 200 metrelik tampon bölgenin içine düşüp düşmediğini görebiliriz.

Katmanların çakıştırılması işlemi, hem vektörel hem de hücresel veri kümeleri ile kullanılabilir. Vektörel veri kümelerinde bu işlem zaman alıcı, karmaşık ve zor hesaplamalar içerirken hücresel veri kümelerinde hızlı ve etkilidir.

2.4.7.1 Vektörel Katman Çakıştırma:

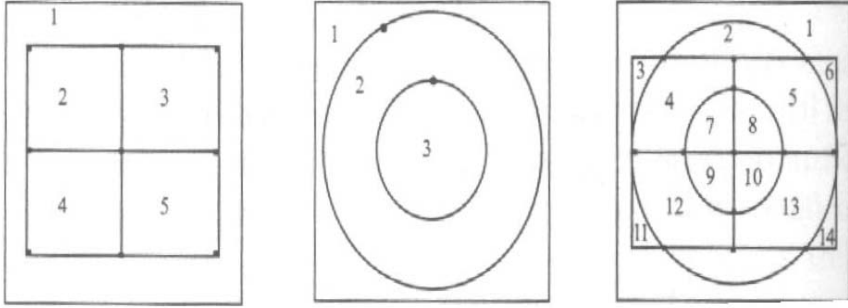
Vektörel katman çakıştırma sonucunda elde edilecek olan vektörel katman, eldeki diğer iki katmana göre daha fazla nesne içereceğinden daha karmaşık olacaktır. Bu işlem sırasında noktaların hangi alanların içine düştüğü, örneğin meteoroloji istasyonlarının hangi arazi kullanım sınıfına denk geldiği, çizgilerin ve alanların hangi alanlara düştüğü gibi vektörel karşılaştırmalar yapılır. Bu noktada eldeki verilerin topolojik

olarak problemsiz olması yani nesneler arasındaki kesişme, içirme, komşu olma gibi bağlantılarda anlamsızlık olmaması gerekmektedir.

Çizgilerin ve alanların birbirleriyle kesişmelerinden yeni çizgi parçaları veya yeni alanlar oluşabilir. Örneğin, katman çakıştırma sonucunda bir yol bir arsayı iki ayrı alana bölebilir. Katman çakıştırma işlemi sırasında alanların çakışması farklı şekillerde yorumlanabilir. Örneğin, iki farklı katmandaki alanların birleşimiyle yeni alanlar oluşabilir veya farklı katmanlardaki alanların sadece kesişim bölgesi yeni veri kümesine aktarılabilir. Şekil 2.13’de ilk iki şekilde verilen vektörel katmanların birleştirilmesiyle üçüncü şekildeki yeni vektörel katman elde edilmiştir.

2.4.7.2 Hücresel Katman Çakıştırma:

Hücresel veri modelinde herşey hücrelerle ifade edildiği için katman çakıştırma işleminde kullanılan yöntemler vektörel veri modelinden biraz farklıdır. Çakıştırılan katmanlardaki üst üste gelen hücrelerin değerleri eklenebilir, çıkarılabilir, çarpılabilir, bölünebilir, hatta farklı matematiksel fonksiyonlardan geçirilebilir. Bunun sonucunda yeni oluşacak katmandaki hücrenin değeri bu uygulanan harita matematiği sayesinde elde edilir. Örneğin üst üste gelen farklı iki katmanın hücrelerinden biri yol olduğu, diğeri ise arazinin ekili alan olduğunu gösterebilir. Bu iki katman çakıştırılınca yeni katmandaki hücrenin değeri, üzerinden yol geçen bir ekili alanı gösterir.

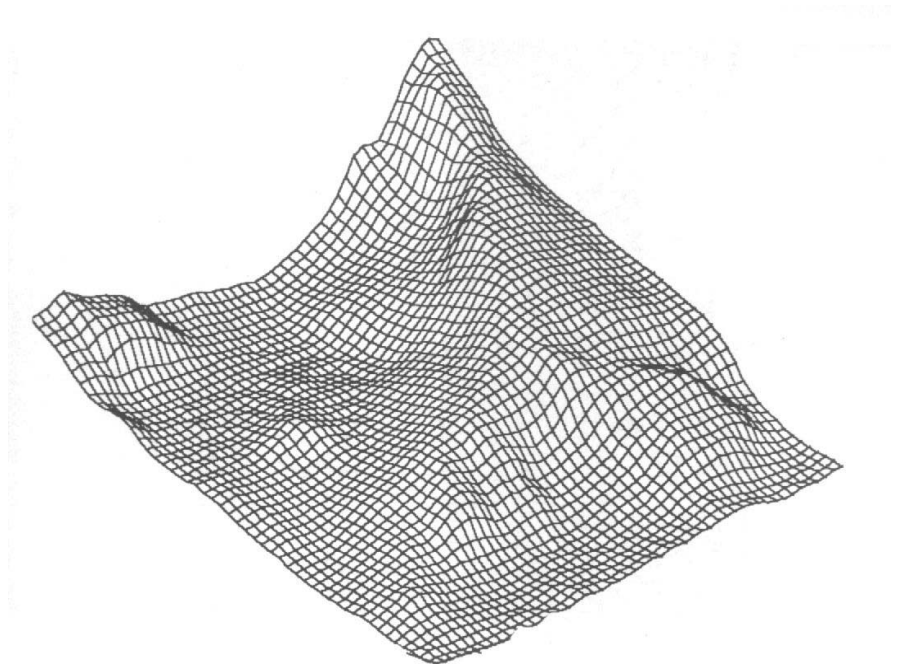


Şekil 2.13: İki vektörel katmanın birleştirilmesi
(Heywood et al., 2002).

2.4.8. Yüzey Analizleri

CBS'deki yüzey analizi tekniklerinin temeli enterpolasyona dayanır. Çünkü burada amaç, genellikle sınırlı sayıdaki veriden sürekli bir yüzey elde etmektir. Yeryüzü üç boyutlu olduğuna göre tüm CBS uygulamaları üç boyutlu analizleri dikkate almak durumundadır. Temel üç boyutlu analizler; eğim (slope), bakı (aspect), kabartma (relief), ışıklandırma (hillshade), profil (profile) ve görünürlük (visibility) olarak sıralanabilir. (Taştan vd., 1994)

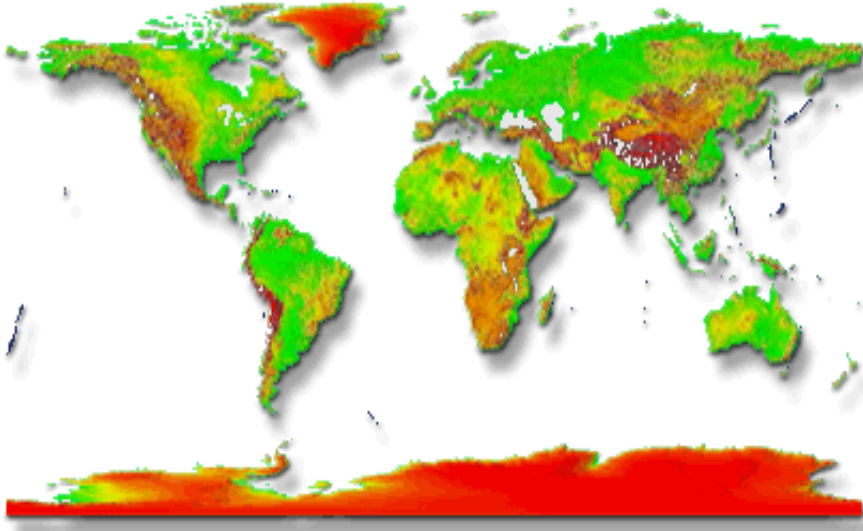
Eğim, birim yüzeydeki diklik olarak tanımlanabilir ve açısal derece veya yüzde olarak ifade edilir. Bakı, birim yüzeyin baktığı yönü tanımlar ve kuzeyle yapılan açı olarak ifade edilir. Kabartma harita ise birim yüzeylerin yükseklik değerlerine göre renklendirilmesiyle elde edilen görüntüdür. Işıklendirma, kabartma haritaya belirli bir yönden ve yükseklikten ışık düşürülmesiyle elde edilen üründür. Şekil 2.14, örnek bir bölgenin tek kafes şeklinde ifade edilmiş yüzey modelini; Şekil 2.15 ise, dünyanın bir kesiminin yükseklik değerlerine göre sunumunu göstermektedir.



Şekil 2.14: Bir bölgenin tel kafes şeklinde çizilmiş yüzey modeli
(Jones, 1999)

Profil ise, üç boyut verisinden faydalanarak yeryüzündeki belirli iki nokta arasındaki çizgide yükseklik değerinin değişimini gösterir ve genellikle grafik şeklinde sunulur. Görünürlük analizi yeryüzündeki belirli bir nokta ve yükseklikten bakıldığında belirli bir uzaklık dahilinde hangi bölgelerin görülebildiği ve hangilerinin görünemediğini ortaya koyan analizdir.

Bu analizler askeri ve mühendislik amaçları için kullanılabileceği gibi, sadece istenilen bölge hakkında genel bir fikir edinmek amacıyla da uygulanabilir. Bölge seçimi projelerinde de yüzey analizleri sıklıkla kullanılır. Çünkü, bölge seçimi için bölgenin eğimi, baktığı yön, nerelerden görüldüğü gibi kriterler önem taşır. (Komesli, 1999)



Şekil 2.15: Dünyanın yüksekliklere göre renklendirilmiş gösterimi
(<http://www.maptell.com>, 2006)

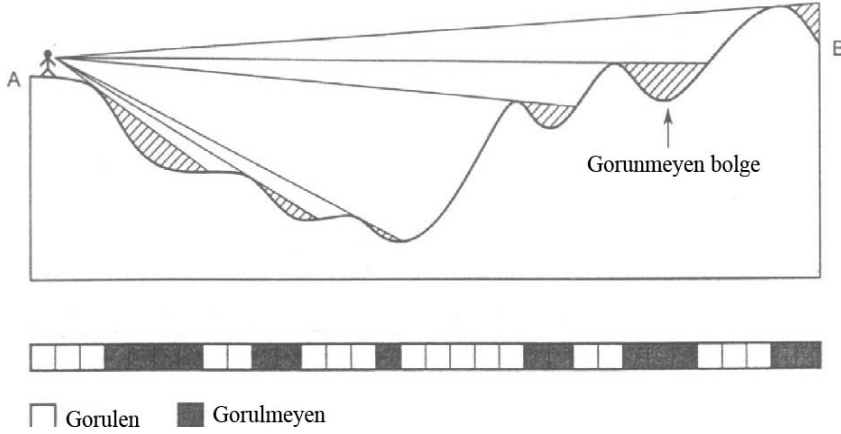
Ayrıca, üç boyut verisi ile istenilen bölgenin perspektif görüntüsü veya üç boyutlu modeli de elde edilebilir. Piyasadaki birçok CBS uygulaması üç boyutlu modellemeyi desteklemektedir. Üç boyutlu model oluşturulduktan sonra bunun içinde dolaşmak, üzerinde uçuş simülasyonu yapmak ve değişik açılardan ve yüksekliklerden incelemek mümkün olmaktadır. Şekil 2.16 bir CBS yazılımı ile çalışma bölgesinin yüzeyinin modellenmiş ve sel baskınının simüle edilmiş durumunu göstermektedir. Şekil 2.17 ise iki nokta arasında görünürlük analizi sonuçlarını içermektedir.



Şekil 2.16: Sel baskın yüzey modellemesi örneği
(<http://www.sbe.hw.ac.uk>, 2006).

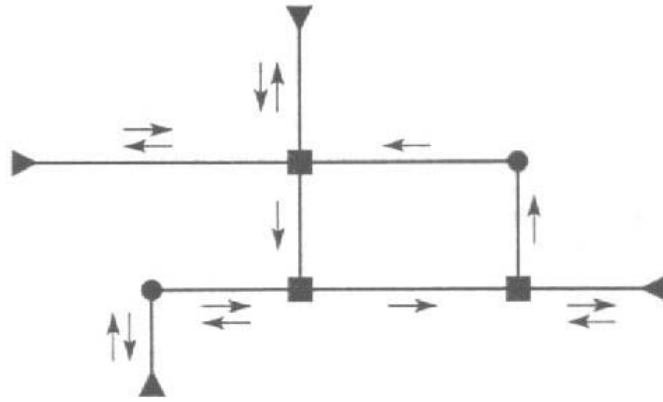
2.4.9. Ağ Analizleri

Ağ, üzerinde kaynakların aktığı birbirine bağlı çizgi kümelerinin oluşturduğu yapıdır. Örneğin, yollar, nehirler, boru hatları ağlara örnek olarak verilebilir. Klasik ağ problemleri en kısa yolu bulma, noktalar arasında seyahat etme, yol izleme olarak verilebilir. Örneğin, yol haritasında iki şehir arasındaki en kısa yol uzunluğunu bulmak en kısa yolu bulmaya, bir satış elemanının dolaşacağı beş şehir için en uygun rotayı belirlemesi noktalar arasında seyahat etmeye, bir kanalizasyon şebekesinde bir evden çıkan atık suyun arıtma tesisine ulaşana kadar izleyeceği yolu bulmayı yol izlemeye örnek olarak verebiliriz. Şekil 2.18, bir ağ yapısında bulunan elemanları örneklemektedir.



Şekil 2.17: A ve B noktaları arasındaki görünürlük analiz sonuçları(Heywood et al., 2002)

En kısa yolu bulma mesafe cinsinden olabileceği gibi herhangi bir masraf cinsinden de olabilir (Hong et al., 1995). Örneğin, bir aracın tüketeceği yakıt cinsinden veya ulaşım için harcanan vakit cinsinden olabilir. Bazen gidilen yol kısa olsa bile ulaşım için harcanan süre veya tüketilen yakıt fazla olabilir. Bazı durumlarda ağ üzerinde her dönüş uygun olmayabilir veya bazı yollar tek yönlü işliyor olabilir. Ağ problemlerini çözmek için tasarlanan algoritma bu koşulları dikkate almalıdır. Noktalar arasında seyahat etmede iki nokta arasında gitmenin alternatifleri olduğu gibi dolaşılacak tüm noktaların sıralamasının da alternatifleri olabilir. Örneğin, bir satış elemanının on müşteriye gitmesi gerekiyorsa, bu müşterileri ziyaret etme sırası da önemli değilse, ağ algoritması $9!$ yani 362,880 alternatifi değerlendirmek durumundadır. Buna bir de ağ yapısındaki gidiş alternatifleri eklenirse problemin ne kadar karmaşık bir hal alacağı görülebilir. Bu tür problemlerin elle çözülmesi genellikle mümkün değildir.



● Kırılma Noktalari

■ Kesisim Noktalari

▲ Bitis Noktalari

⇔ Cift Yon

→ Tek Yon

Şekil 2.18: Ağ örnek gösterimi (Heywood et al., 2002).

3. ANLAMSAL WEB VE ONTOLOJİ

Şubat 2001 tarihinde, “World Wide Web Konsorsyumu (W3C)” tarafından, web üzerinde verinin anlamsal gösterimini sağlamak maksadıyla “Anlamsal Web (Semantic Web)” kavramı ortaya atılmıştır. World Wide Web’in kurucusu Tim Berners Lee, Anlamsal Web’in şimdiki web teknolojisinin bir uzantısı olduğunu, ancak bilginin anlamının daha iyi bir şekilde tanımlandığını, böylelikle bilgisayar ve kullanıcıların daha iyi bir şekilde birlikte çalışabilirliklerinin sağlandığını ifade etmiştir. Web dokümanlarının içeriği artık sadece insanlar için değil, aynı zamanda bilgisayarlar için de anlam ifade eder hale gelmektedir. Anlamsal web’in çalışabilmesi için, bilgisayarlar yapısal (structured) bilgiye ve çıkarsama kurallarına (inference rules) ihtiyaç duyarlar. İşte bunu başarmanın yolu da, ontoloji kullanmaktan geçer. Ontoloji, kısaca bir “kavramsallaştırma tanımlanması”dır, “yapısal bir sözlük” ve “anlamsallık” sağlar. Ontolojilerin yardımıyla, bilgisayarların veriyi anlamasını sağlamak için web kaynakları işaretlenir. (Doerr, 2003)

3.1. Anlamsal Web’in Tanımı

Anlamsal Web; verinin uygulamalar, kuruluşlar ve topluluklar arasında paylaşılmasına ve yeniden kullanılmasına imkan veren ortak bir çalışma ortamı sunmaktadır. Çok sayıda araştırmacı ve endüstriyel katılımcılardan oluşan W3C Konsorsyumu tarafından yönetilen ortak bir çalışmadır. Anlamsal web, XML’in bir üst sunumu olan RDF teknolojisine dayanmaktadır.

Anlamsal web’i kısaca tanımlamak gerekirse, verilerin web’i denilebilir (<http://www.w3.org/2001/sw/>, 2006). Günlük hayatımız esnasında kullandığımız, fakat web’in bir parçası olmayan birçok veri

vardır. Web üzerinde banka hesaplarımızı, fotoğraflarımızı ve bir takvim üzerinde işaretli randevularımızı görebiliriz.

Günümüz teknolojisinde veri, uygulamalar tarafından kontrol edilmekte ve her uygulama veriyi kendisi için saklamaktadır.

Anlamsal web iki konu ile yakından ilgilidir: 1) Verinin değişimi için ortak formatlar (web üzerinde normal olarak sadece dokümanların değişimi olmaktadır), 2) verinin gerçek dünya nesneleri ile nasıl ilgili olduğunun ortaya koyulduğunu kaydeden bir programlama dili. Bunların sayesinde kullanıcı ve bilgisayarlar bir veritabanından başlayıp sonsuz sayıda veritabanında dolaşabilirler.

Anlamsal Web'in arkasındaki temel fikir, insana özgü web faaliyetlerini bilgisayarlara yaptırabilmeyi sağlamaktır. Çünkü, bilgisayarlar yapabildikleri herşeyi, insanlardan hem daha iyi, hem de daha hızlı yaparlar. Ancak, bilgisayarların bu yeni görevlerini yerine getirebilmeleri için, insanların web'deki veriyi tümüyle otomatik olan işlemlere uygun ve makineler tarafından okunabilir hale getirilmesi gerekmektedir. Bunu başarmak ise, daha fazla üst veri (veri hakkında veri) oluşturmak anlamına gelmektedir. Bu amaçla XML tabanlı RDF vb. teknolojiler kullanılabilmektedir. Bu durumda bilgisayarlar, sadece verinin ne anlama geldiğini değil, aynı zamanda web üzerinde başka yerlerde bulunan çeşitli veri parçalarının da aynı anlama gelip gelmediğini bilmek zorundadırlar (aynı görünmeseler bile). Örneğin web üzerinde "ad" ve "isim" olarak tanımlı farklı iki verinin anlamsal olarak aynılık derecesi tanımlı olmalı ve bu bilgisayarlar tarafından algılanabilmelidir (Kiyoki, 1995).

Anlamsal web'in yardımıyla anlamsal olarak zengin ve tanımlayıcı bilgiler, herhangi bir kaynak (resource) ile ilişkilendirilebilir. Örneğin, doküman yaratılması hakkında üstveri ilave ederek Rıfat Ilgaz'ın bir

“yazar” olduğunu belirten üstveri içeren dokümanlar aranabilir. Biraz daha fazla üstveri yardımıyla ise, sadece “araştırma makaleleri” kategorisi altındaki dokümanlar aranabilir. Anlamsal web yardımıyla, geçmişte olduğu gibi sadece dokümanlar için değil aynı zamanda insanlar, kavramlar ve ilişkiler için URI tanımlanmaktadır. Örneğimizi ilerletecek olursak, kişiye “yazar” rolünde ve “araştırma makaleleri” kavramında belirleyiciler vererek kişiyi ve karşılık gelen ilişki daha belirli hale getirebiliriz.

Bütün bu bilgilerden sonra Anlamsal web’in tanımını yapmak gerekirse: “Anlamsal Web, web’deki unsurlar arasında ilişkiler tanımlayan ve bilgisayarlar tarafından otomatik işleyebilme gücü sağlamak amacıyla anlamsal bilgi içeren dokümanlar veya doküman parçalarını içeren web’dir”. (Koivunen and Miller, 2001)

3.2. Anlamsal Web’in Temel Prensipleri

Prensip 1: Her şey URI’ler ile tanımlanabilir.

Fiziksel yeryüzü üzerindeki insanlar, yerler ve nesneler; çeşitli tanımlayıcılar (identifier) kullanılarak referans edilebilirler. Web üzerinde bunları tanımlayan URI’ler kolaylıkla veya dolaylı olarak yaratılabilir. Örneğin, Ankara Belediyesi tarafından yönetilen ve Ankara şehri hakkında bilgileri içeren bir sayfanın URI’si gösterilerek Ankara şehri gibi bir yeri tanımlamak mümkündür. Daha çok üstveri yardımıyla gerçek bir şehir ile URI arasındaki ilişki daha detaylı olarak tanımlanabilir.

Prensip 2: Kaynak ve bağlantıların tipleri olabilir.

Günümüzde web, kaynaklar (resource) ve bağlantılardan (link) oluşmaktadır. Kaynaklar, insanların kullanımını amaçlayan web dokümanlarıdır. Dokümanlar, genellikle kaynakların ne için olduğunu ve diğer web dokümanları ile ilişkilerinin neler olduğunu açıklayan

üstverileri içermeyiz. Bir web sayfasının araştırma yazısı mı olduğu, yoksa bir roman hakkında mı bilgi verdiği insanlar tarafından kolaylıkla algılanabilirken, bu bilgilerin bilgisayarlar tarafından algılanması olası değildir.

Anlamsal web’de, bilgisayarlara konsepti biraz da olsa verebilen kaynak ve bağlantıların tipleri mevcuttur. Örneğin, bazı bağlantılar bir kaynağın diğer bir kaynağın versiyonu olduğu veya bir kişiyi tanımlayan bir kaynak tarafından yazıldığını ifade edebilir. Bir diğer örnek ise, bir kaynak diğer bir yazılıma bağlı olan bir yazılım içerebilir.

Prensip 3: Tam olmayan bilgiler kabul edilebilir.

Günümüzde yaygın olarak kullanılmakta olan web, ilişkisizdir(unbound). Diğer bir deyişle, ölçeklenirlik uğruna bağlantıların bütünlüğü göz ardı edilebilmektedir. Web sayfaları geliştirilirken bağlantı konulan web sayfaların, yine kendi web sayfasına bağlantı içermesi mümkündür. Aynı zamanda, bağlantı konulan web kaynağı bir şekilde kaldırılmış veya silinmiş olabilmektedir.

Bu gerçeğe anlamsal web tarafından da bakılacak olursa, anlamsal web de ilişkisizdir. Anlamsal web üzerinde, herhangi bir uygulama geliştiricisi istediği bir konu hakkında istediği gibi web sayfasını geliştirebileceği gibi, kaynaklar arasında da farklı tiplerde bağlantılar yaratabilmektedir. Anlamsal web’de her zaman için keşfedilecek daha başka kaynaklar mevcuttur. Bazı bağlantı kurulmuş kaynakların mevcudiyeti sona ermiş olabilmekle beraber, web adresleri tekrar kullanılabilir. Anlamsal web araçları, verinin bozulmasını göz ardı edebilmekte ve buna rağmen işlemeye devam edebilmektedirler. Örneğin, Tim Berners Lee’nin, W3C faaliyetlerindeki rolünü öğrenebilmekte ve Lee’nin diğer çalışmalarını içeren bağlantılar kayıp olsa da bu bilgiyi çıkarsama yapma maksadıyla kullanılabilir.

Anlamsal web araçları, genelde seçilen bilgi adalarında faaliyet göstermektedir.

Prensip 4: Tam gerçeğin bilinmesine ihtiyaç yoktur.

Web üzerinde bulunan her bilgi doğru değildir. Anlamsal web de bu gerçeği değiştirmemektedir. Gerçek veya güvenilirlik web üzerinde her uygulama tarafından değerlendirilmekte ve işlenmektedir. Uygulamalar, sözcüklerin bağlamını (context of statement) kullanarak hangi bilginin güvenilir olduğuna karar verebilirler. Örneğin, Kim? Ne zaman? Ne söyledi? gibi sorular anlamsal web’de kolay bir şekilde cevap bulabilecektir.

Prensip 5: Gelişme desteklenmektedir.

Benzer konseptlerin farklı kişilerce farklı yerlerde veya aynı kişilerce farklı zamanlarda tanımlanmakta olduğu bir gerçektir. Benzer konseptleri kullanan web üzerindeki bilgileri birleştirmek çok faydalı olacaktır. Anlamsal web, insan zekasının anlayabildiği üzerinde uzlaşmış tanımlayıcılar (descriptive conventions) kullanmaktadır. Bu tanımlayıcılar, farklı grupların çalışmalarını, farklı sözlükler kullansalar bile etkili olmasına imkan vermektedir. Anlamsal web, çalışma gruplarına tutarsızlıkları belirlemeleri ve çözmeleri için kullanabilecekleri araçlar sağlamaktadır. Aynı zamanda, yeni bilgiler eskilerinin değiştirilmesine gerek kalmadan eklenebilmektedir.

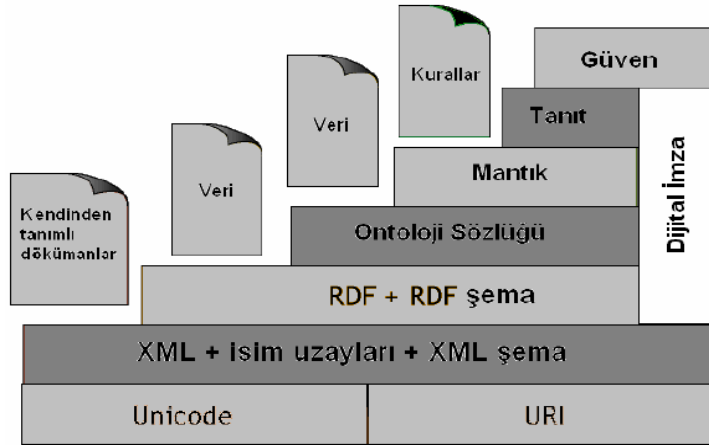
Prensip 6: Minimum tasarım.

Anlamsal web, basit olan şeyleri daha basit ve karmaşık olanları ise mümkün hale getirmektedir. Kısaca belirtmek gerekirse W3C’nin amacı, gerekli olan nesneleri standartlaştırmaktır. Bu yaklaşım, standart teknolojilere dayalı uygulamaları geliştirme imkanı vermektedir. Aynı zamanda, gelecekte karşılaşılabilecek karmaşıklığı çözmek amacıyla planlar da mevcuttur. Anlamsal web teknolojileri kullanılması

sonucunda, sadece tam olmayan parçacıkların birleşimi yapılmış olmayacaktır.

3.3. Anlamsal Web'in Katmanları

Yukarıda bahsedilen Anlamsal Web prensipleri, web teknolojileri ve standartlarını, katmanlar şeklinde uygulamaktadır. Tim Berners Lee'nin Anlamsal Web için sunduğu katmanlı mimari Şekil 3.1'de görülmektedir (Berners Lee, 1999).



Şekil 3.1: Anlamsal Web Mimarisi (<http://james.seng.sg>, 2006)

Unicode ve URI katmanları uluslararası karakter kümeleri kullanılmasını sağlarlar ve Anlamsal Web'de nesnelerin tanımlanabilmesi için imkan sunarlar. XML katmanı, isim uzayları ve şema tanımları ile birlikte, anlamsal web tanımlarının diğer XML tabanlı standartlar ile bütünleştirilmesini sağlar (<http://www.w3.org/XML>, 2006). RDF ve RDFS ile URI'ler ile nesneler hakkında ifadeler (statement) yapılmasına ve URI'ler tarafından ilişkilendirilen sözlükler hazırlanmasına olanak sağlarlar. Bağlantı ve kaynaklara tiplerin verildiği yer, bu katmandır. Ontoloji katmanı, sözlüklerin değişimini sağlar ve farklı kavramlar arasındaki ilişkileri tanımlar. Dokümanlardaki

değişiklikler ise, sayısal imza katmanı tarafından belirlenir. Günümüzde W3C çalışma grupları tarafından bu katmanlar standart haline getirilmektedir.

Üst katmanlar olan Mantık, Tanıt ve Güven, halen araştırma aşamasında olup, basit uygulama gösterileri geliştirilmektedir. Mantık katmanı kuralların yazılmasını sağlar. Tanıt katmanı kuralları uygular ve Güven tabakasının mekanizması ile birlikte verilen tanıtlara güvenilir olup olmadığı değerlendirilir.

3.4. Ontoloji

Ontoloji, sözlük anlamı olarak “varolma bilimi” diye tanımlanmaktadır. Gruber, günümüzde çoğunlukla kabul gördüğü üzere, ontolojiyi “kavramlaştırmanın kesin tanımı” olarak tarif etmiştir (Gruber, 1993a, 1993b). Olası bir ontolojinin en basit tanımlarından birisi de “kontrol edilebilir bir sözlük” olabilir. Buna örnek olarak kısıtlı sayıdaki terimlerin bir listesi gösterilebilir. Ayrıca, katologlar da terimlerin belirli anlamlarını ifade ettiklerinden dolayı, bu tanıma örnek teşkil edebilirler. Sözlükler de aynı zamanda, kullanıcıların anlayabildikleri bir dilde terimlerin basitçe anlamlarını ifade ettiklerinden dolayı bir anlamsallık (semantics) içermektedirler. Fakat, sözlüklerdeki izah edici tanımlar, belirli olmadıkları için etmenler tarafından uygun olarak algılanamamakta ve dolayısıyla bilgisayar ortamında işlenilebilirliği olmamaktadır.

3.4.1. Ontoloji Özellikleri

Bir tanımlı ontoloji olarak adlandırmak için aşağıdaki özelliklerin mevcut olması gerekmektedir (McGuinness, 2002) :

- sınırlı kontrollü sözlükler
- belirli anlamdaki terim ve sınıf ilişkileri
- sınıflar arasındaki hiyerarşik alt sınıf ilişkileri

Bu özelliklere sahip olan tanımlar, basit ontolojiler diye adlandırılmaktadır. Ayrıca, zorunlu olmayan tipik özellikler ise şunlardır:

- her sınıf için özellik tanımlamaları
- ontolojiye münferit olarak dahil olma
- her sınıf için değer kısıtlama tanımı

Son olarak, olması arzulanan, fakat zorunlu ve tipik olmayan özellikler ise şunlardır:

- ayırık sınıfların tanımlanması
- terimler arası mantıksal ilişkilerin tanımlanması.

3.4.2. Anlamsal Web ve Ontoloji

Farklı veri tabanlarında, aynı kavram için farklı tanımlar kullanılmaktadır. Anlamsal web`de ontolojiler kullanılarak bu tür problemler çözülmeye çalışılmaktadır. Bir anlamsal web dökümanı, terimlerinin anlamlarını ve bu terimler arası ilişkileri ifade etmek için bir ontolojiye işaret etmektedir.

Anlamsal web, yapay zeka araştırmacılarının itina ile hazırladıkları ontolojilerden ibaret değildir. Büyük sayıdaki kullanıcıların paylaştıkları karmaşık ve büyük ontolojiler, yerlerini, küçük, kendine özgü ve birbirleri ile ilişkilendirilmiş ontoloji parçacıklarına bırakacaklardır. Yakın gelecekte web kullanıcıları web içeriklerini bu parçacıkları kullanarak geliştireceklerdir (Hendler, 2001).

Gelecek birkaç yıl içerisinde, hemen hemen bütün üniversite, şirket ve devlet kurumları kendi web kaynaklarını bir ontolojiye ilişkilendireceği değerlendirilmekte ve bunun sonucunda da veri tabanı gibi çeşitli uygulamaların serbestçe web üzerinden paylaşımının sağlanacağı varsayılmaktadır.

3.4.3. RDF ve RDFS

RDF, önceleri web tabanlı kaynakların üst verilerini açıklamak ve tanımlamalarını standart hale getirmek amacıyla W3C tarafından tavsiye niteliğinde sunulmuştur. RDF, üst veri olsun veya olmasın her türlü verilerin gösterimi için birçok kolaylık sağlamaktadır.

RDF'deki temel yapı taşı, çoğunlukla P(S,O) şeklindeki “Subject-Predicate-Object” üçlüsüdür. Bu, “S öznesinin O değerine sahip bir özelliği (P) var.” anlamı taşımaktadır. Diğer bir şekilde açıklamak gerekirse; iki düğüm arasında etiketlenmiş bir kenar olarak ilişkilendirilebilir. Bu notasyon oldukça faydalıdır. Çünkü, RDF kullanarak özne ve nesnelerin birbiri arasında değiştirilebilmesi mümkün olmaktadır. Şöyle ki, bir üçlüdeki herhangi bir özne, diğer bir üçlüdeki nesne rolünü üzerine alabilir. RDF, aynı zamanda herhangi bir RDF durumunun (statement) bir üçlü özne veya nesnesi olabileceğini somut bir şekilde ortaya koymaktadır. Bu da, çizgelerin birbirine bağlı olduğu gibi içiçe (nested) de olabileceği anlamını taşımaktadır. Böylelikle, web üzerinde başkalarının yarattığı durum (statement) hakkında olumlu veya olumsuz fikirlerin açıklanmasına imkan verilecektir. (Broekstra et al., 2002)

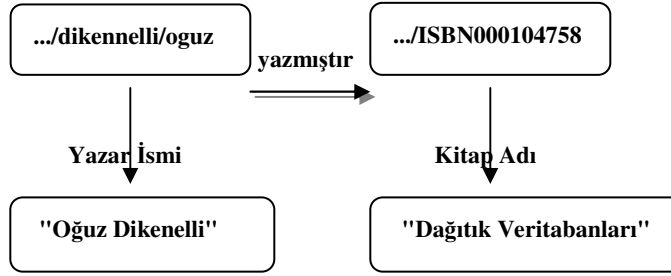
RDF model ve sözdizimi, aynı zamanda RDF veri modelleri için bir XML sözdizimi'ni sunmaktadır. Bu durum Şekil 3.2'de açıklanmıştır. Şöyle ki,

```
<rdf:Description rdf:about=
"http://www.unluyazarlar.com/dikenelli/oguz">
  <s:ismi>Oguz Dikenelli</s:ismi>
  <s:yazmistur> rdf:resource="http://www.kitaplar.org/ISBN0001047582"/>
</rdf:Description>
<rdf:Description rdf:about=
```

```

"http://www.kitaplar.org/ISBN0001047582">
  <s:başlık>Dağıtık Veri Tabanları</s:başlık>
</rdf:Description>

```



Şekil 3.2: Bir RDF Çizge Örneği (Komesli vd., 2002)

Dikkat edileceği gibi RDF, web verileri için temel bir “subject-predicate-object” modeli sağlamak için tasarlanmıştır. Bundan başka bir veri modelleme amacı gütmemekte, özellikle daha ileri seviyede veri modellemesi için daha ayrıntılı tanımlar sunmamaktadır.

RDFS, sistem geliştiricilerinin RDF verileri için (“yazmıştır” ifadesi gibi) özel bir sözlük tanımlamalarına imkan veren ve kendisine ifadelerin uygulanabildiği (“yazar” sınıfı gibi) çeşitli nesneler belirleyen bir mekanizmadır. RDFS, bu işleri daha önceden “class”, “subclass”, ve “property” gibi terimleri tanımlayarak ve bu terimleri sonradan uygulamaya dönük şemalarda kullanarak yerine getirmektedir. RDFS tanımları, aynı zamanda geçerli RDF tanımlarıdır. Aslında, normal RDF tanımlarından tek farkı; RDFS’de bir ifade belirli terimlerin anlamı (semantics) ve tercümesi olarak yapılmaktadır. Örneğin, “subclassOf” özelliği, sistem geliştiricilerine sınıfların hiyerarşik kuruluşunu belirlemelerine imkan vermektedir. Böylelikle, nesneler, “type” özelliği kullanarak ifade edilebilmektedirler. Özelliklerin kullanımı üzerindeki kısıtlamalar, “domain” ve “range” yapıları kullanılarak belirlenmektedir.

4. COĞRAFİ ANLAMSAK WEB (GEO SEMANTIC WEB)

Coğrafya, günlük hayatımızda meşgul olduğumuz her faaliyet ile doğrudan ilgilidir. Coğrafya, insanların iş hayatlarında günlük olarak yapmış oldukları faaliyetlerinden; yangın, sel, deprem gibi doğal afetlere kadar, zamana ve duruma bağı (spatio-temporal) şekilde coğrafi olarak referans edilebilen bir dizi faaliyetleri analiz edebilmek için temel bir çalışma ortamı sunar. Bu karışıma “Anlamsal Web” kavramını da ekleyerek, bazı yeni, yararlı ve kullanışlı uygulamalar elde edilebilir.

Coğrafik referanslar gerçek dünya bilgilerini açıklamada faydalıdırlar. Bazı coğrafik referanslar, çok iyi bilinen yerler ile ilgili coğrafik ilişkileri (bir havaalanından bir hotele olan uzaklık, cadde isim ve yönlerini içeren tarif bilgileri vb.) içermekte ve çok sık bir şekilde günümüz web teknolojisinde kullanılmaktadırlar. Coğrafik referansların insanlar için anlaşılması çoğu durumlarda kolaydır. Fakat, düz metin biçiminde (plain-text format) oldukları için anlamsal karmaşıklığın çözülebilmesine ihtiyaç vardır. Bunun için ilgili bilgiye (knowledge) ihtiyaç duyulmaktadır. Bu da coğrafi referansların bilgisayarlar tarafından anlaşılmasını zorlaştırmaktadır. Eğer, bilgisayarlar sözü geçen coğrafi referansları, web’deki “hyperlink”ler gibi esnek olarak işleyebilseler, harita üzerinde simge yerleştirmenin yanında, açıklayıcı coğrafi bilgileri de yaratabileceklerdir. Aslında bu durum, **ilgili ontolojiye dayalı olarak yeni yöntemlerle bilginin otomatik olarak kullanılmasını sağlayan Anlamsal Web’den beklenen şeydir**. Örneğin, Anlamsal Web sayesinde web sayfalarındaki coğrafik referansları birbirine bağlayarak ve insanların düşünmekten ziyade sezgi yolu ile öğrendiği referansları içeren bilgileri kullanarak bir seyahat rehberi oluşturmak mümkün olabilecektir (Harvey, 1999).

Günümüzde web üzerinde mekansal verilerin bulunması, bilgisayar kullanıcıları için oldukça zor, akıllı sayısal ajanlar için ise hemen hemen imkansızdır. Bu sebepten dolayı, mekansal veri portalları ortaya çıkmaya başlamıştır. (Hernandez et al., 1995) Gelecekte anlamsal web sayesinde bilgi üreten akıllı etmenler daha çok kullanıma girecektir. Bunun sonucunda ise, mekânsal karar verme daha kolay mümkün olabilecektir. Mekânsal Karar Destek Sistemi (Spatial Decision Support System-SDSS)'nde mekânsal karar verme işleminde tipik olarak heterojen mekânsal ve mekânsal olmayan veriler birleştirilir. Aynı zamanda SDSS, farklı tiplerdeki bilgilerin bütünleştirilmesini gerektirir. CBS'lerde, coğrafi bilginin anlamsal içeriğine dayalı olarak, fakat gösterim şekli ile bağlantısız şekilde diğer bilgilerle pürüzsüz (dikişsiz-seamless) bütünleştirme gerçekleştirilmektedir. Anlamsal olarak birlikte çalışılabilirlik (Semantic Interoperability) düzgün olarak tanımlanmıştır. Bunun yanı sıra ontolojiler içerisinde açıklanabilen kavram ve terimlere de ihtiyaç duyulmaktadır. (Bishr, 1999)

4.1 Coğrafi İşaretleme Dili (Geographic Markup Language - GML)

Coğrafi İşaretleme Dili (Geographic Markup Language – GML), coğrafi bilgilerin modellenmesi, taşınması ve depolanması için XML Şema kullanılarak yazılan bir XML grameridir.

Yeryüzünü modellemek için GML tarafından kullanılan temel konsept, OGC'nin tanımlamalarından (Abstract Specification) elde edilmiştir. (<http://www.opengis.org/techno/abstract.htm>, 2006)

GML; arazi arızalarını, koordinat referans sistemi, geometri, topoloji, zaman, ölçüm birimleri ve genellenmiş değerleri içeren coğrafyayı tanımlayan birçok çeşitte nesneler sağlar.

Bir coğrafik detay, gerçek dünya olgusunun bir soyutlandırılmasıdır. Kısaca, bir nesne şayet dünya üzerinde bir yer ile ilişkilendiriliyorsa, o nesnenin bir coğrafik detay olduğu söylenebilir. Dolayısıyla, yeryüzünün sayısal gösterimi, bir detaylar kümesi olarak düşünülebilir. Detayın durumu, her özelliğin bir {isim, tip, değer} üçlüsü olarak düşünülebildiği bir özellikler kümesi (set of properties) tarafından tanımlanabilir.

Tip ve isimleri ile birlikte, bir detayın sahip olabileceği özelliklerin sayısı, onun tip tanımlaması ile belirlenir. Coğrafi detaylar, geometri içeren ve geometri-değerli özellikleri olan bir örnektir. Bir detay kümesi, kendisinin de bir detay olarak tanımlanabileceği detayların toplamıdır. Sonuç olarak, bir detay kümesinin bir detay tipi vardır. Dolayısıyla içerdği detaylara ilave olarak, kendisine ait belirgin özellikleri de (distinct property) mevcut olabilir.

GML’de coğrafik detayların örtü (coverage) ve gözlem (observations) olarak iki adet alt tipi vardır.

Bir örtü; bir mekansal etki alanı ve 2’den n’e kadar homojen değer kümesi ikilileri (tuple) ile birlikte bir örtü fonksiyonu oluşturan detayların alt-tipidir. Diğer bir deyişle bir örtü, bir detay veya “yeryüzü olgusunun mekansal dağılımı, mekansal ilişkilerinin görülebilir hali ve modellenmesini gösteren” bir detaylar kümesidir.

Bir gözlem (observation), çoğunlukla bir kamera, bir şahıs veya bir çeşit alet (aletler yardımıyla bir gerçeği veya oluşumu farkına varma veya tanımlama hareketi) ile gözlemleme hareketini modeller. Bir gözlem, gözlemin yapıldığı bir zaman ve gözlem için atanan bir değer ile GML detayı olarak tanımlanabilir.

Bir referans sistemi, normal olarak bir ölçüm aralığı belirler. Belirlenen bu ölçüm aralığı içerisinde, bir yere, zamana ve diğer tanımlayıcı miktar ve niceliğe değerler atanabilir.

Bir koordinat referans sistemi ise, koordinat sistemi eksenlerini içermektedir. Bu eksenler yeryüzünün şekli ve büyüklüğünün tanımlandığı bir datum yardımıyla doğrudan yeryüzü ile ilişkilidir.

Datum, herhangi bir noktanın yatay ve düşey konumunu tanımlamak için başlangıç alınan referans yüzeyidir. Diğer bir deyişle, datum, Yer'in şeklini ve boyutunu tanımlayan bir referans sistemidir. İki çeşidi vardır:

Yatay datum: Koordinatlar için referans alınan başlangıç yüzeyine denir.

Düşey datum: Yükseklikler için referans alınan başlangıç yüzeyi
Bir datum; elipsoidi, enlem-boylam oryantasyonu ve fiziksel bir orijin ile tanımlar.

Kısaca ifade etmek gerekirse,

Geoid ve Elipsoid --> Yer'i modellemek için,

Datum --> Modeli ölçmek için,

Koordinat Sistemleri --> Model üzerinde konum belirlemek için,

Harita Projeksiyonları --> Modelin 3-boyuttan 2-boyuta geçişi için kullanılırlar.

GML'deki geometriler, içerisinde ölçüm yapılmasına imkan veren koordinat referans sistemini gösterir. Bir geometrik “complex” veya geometrik küme (aggregate) nin “Parent” geometrisi, bileşen geometriler için bu belirtiyi yapar.

Zamansal referans sistemi, zaman ölçümü ve zamansal uzunluk veya süre için standart birimler üretir. ISO 8601'den sonra, GML'de zamansal referans sistemi olarak UTC ile birlikte Gregoryan takvim kullanılmıştır.

Ölçüm birimleri sözlüğü, uzunluk, ısı, basınç ve birbirleri arasındaki dönüşümlerin fiziksel miktarlarının sayısal ölçüm tanımlarını sağlamaktadır.

4.2 GML'in Kapsamı

GML, coğrafik detayların hem mekansal hem de mekansal olmayan özelliklerini içeren coğrafik bilgilerin modellenmesi, taşınması ve depolanmasına yarayan XML tabanlı bir kodlamadır.

Bu tanım, öyle bir XML Şema sözdizimi, mekanizması ve kuralı sunmaktadır ki; bunlar;

- Coğrafi mekansal uygulama şemaları ve nesnelerinin tanımları için açık ve herhangi bir kuruma ait olmayan bir çalışma çerçevesi sunar,
- GML çalışma çerçevesinin tanımlayıcı yeteneklerinin uygun alt kümelerini destekleyen profillere imkan verir,
- Bilgi toplulukları ve uzman gruplar için coğrafi mekansal uygulama şema tanımlarını destekler,
- Birbirine bağlı coğrafi mekansal uygulama şema ve veri setlerinin bakımı ve yaratılmasına imkan verir,
- Veri kümeleri ve uygulama şemalarının taşınması ve depolanmasını destekler,
- Kurumların tanımladıkları coğrafik bilgi ve uygulama şemalarının paylaşım kabiliyetlerini artırır.

Uygulama yazarların coğrafik bilgi ve uygulama şemalarını GML formatında depolanmasına imkan verir, veya istek halinde diğer saklama formatlarından GML'e çevrilip, sadece şema ve veri taşınması amacıyla da GML kullanabilmektedir.

GML modeli, OGC'nin tanımına ve normatif referans olarak listelenen ISO 19100 standartlarına uygundur.

GML'de uygunluk, test metodolojisi, konsept, çerçeve ve kriterler; ISO 19105: Coğrafi Bilgi-Uygunluk ve Test standartında belirlenmiştir. Bunlar aynı zamanda OGC'nin arayüz standartlarıdır. Bu standartlara uygunluk sağlanması için, bir yazılım uygulaması geliştirilmelidir.

4.3. Coğrafi Ontoloji Dili Özellikleri

Bir coğrafi ontoloji gösterim dilinde, temel olarak dilin temel fonksiyonel yeterliliğinin olmasının yanı sıra gerekli özellikler şunlardır (Fonseca et al., 1999, 2002):

- Yeryüzündeki gerçek coğrafik kavramların gösterimi
- Her coğrafik kavram için veri özelliklerinin (data properties) gösterimi
- Kavramlar arası ilişkilerin gösterimi
- Özelleştirme/genelleştirme kavram hiyerarşilerinin gösterimi
- Diğer kavramlar kümesinden oluşan kavramlar gibi basit birleşim hiyerarşilerinin gösterimi

Diğer taraftan bir dilin aşağıdaki fonksiyonları sağlaması istenebilir:

- Kavramların veri özellikleri (data properties) üzerindeki kısıtlamalarının gösterimi
- İlişkilerdeki kısıtlamaların gösterimi

- Tek varlıklar (individuals-single entity) üzerinde bütünlük kurallarının (integrity rules) ifade edilmesi
- Farklı kavramlara ait tek varlıklar (individuals-single entity) üzerinde bütünlük kurallarının (integrity rules) ifade edilmesi
- Bir karayolunun 10 km. çevresindeki bütün evlerin sınıfı (class) gibi ileri düzeyde bileşim hiyerarşilerinin gösterimi

4.4. DAML+OIL

DAML+OIL, gerçek dünya olgusunun kavramsallaştırılması için oldukça zengin ve açıklayıcı bir dil özelliğine sahiptir. Coğrafi etki alanının (domain) gösterimi için, dilin bilinen zengin özellikleri kullanılabilir. Coğrafi etki alanının karakteristik özelliği, çok fazla büyük veri kümelerinin ve bu veri kümeleri ile ilgili gösterilecek bilginin mevcut olmasıdır. Bir diğer özel ilgi alanı ise, mekânsal verinin açık veya gizli olarak muhafaza edilmesidir. Tam olarak DAML+OIL dili kullanılarak geliştirilmiş bir coğrafi veri ontolojisinin boyutları oldukça büyüktür. Burada önemli bir soru akla gelebilir: “Açık olarak ne tür bilgiler saklanmalıdır ?” Bu sorunun cevaplanmasındaki zorluk, hesap karmaşıklığının çözümü ile depolama maliyeti arasında verilecek karardadır. Sorun, aynı zamanda kullanılan dilin, mekansal çıkarsama teknikleri kullanarak saklı bilgilerin otomatik olarak ortaya çıkarması yeteneği ile de ilgilidir. Yine, aynı zamanda dilin ölçeklenebilirliği (scalability), gösterim kabiliyeti ve muhtemel binlerce tek varlık (individual) ile ilgilidir. (Denker et al., 2001)

En basit seviyede DAML+OIL, çok iyi bir şekilde yerleştirilmiş anlamsallığı ile birlikte, yeryüzü kavramlarını ifade etmek için çok güçlü bir yöntem sunmaktadır. Kullanıcı tanımlı terminolojiler, tasarımcıların kendi ihtiyaçlarını karşılayacak ontoloji yaratmalarına imkan verir.

Kavramlar;

- sınıflar,
- genelleştirme ve özelleştirme hiyerarşilerin gösterimi için kalıtım mekanizmaları,
- sınıflar veya özellikler arasında bağlantılar oluşturmak için ikili ilişkiler,
- ilişkileri sınırlamak için nitelik çokluğu (cardinality) kısıtlamaları ve
- tanımlanmış sınıfların yaratılmasına olanak vermeye yarayan sınıf yapıcılar (*class constructors*) tarafından elde edilebilirler. Aynı zamanda, yakınlık (*adjacency*) gibi basit mekansal ilişkilerin tanımlanmasında kullanılan *transitive* ve *symmetric* özelliklerinin desteği vardır. Bununla beraber, coğrafik ontolojinin gösteriminde hangi dilin kullanılacağı konusunda bir çok sorun çıkmaktadır. Bu sorunlar, dillerin tasarlanan özelliklerine uyumlu olmalarına engel olmaktadır. Belli başlı temel iki sorun aşağıda açıklanmıştır:

Belli başlı sorunlardan birisi, DAML+OIL'in kendi veri tiplerini (*Data Types*) desteklememesidir. Bunun yerine, XML veri tiplerini kullanmakta, bu da iki veri değeri arasında mantıksal karşılaştırma yapma imkanı vermemektedir. Bunun sonucunda, koordinat verileri üzerinde iki MBR (*Minimum Bounding Rectangle- Enaz Sınır Dikdörtgeni*) arasındaki “içerir (*contains*)” ilişkisinin varlığını test etmek gibi mekansal çıkarsama yapılması zorlaşmaktadır.

Diğer bir sorun ise, araç (*tool*) desteği olmamasıdır. Araçlar, alan uzmanlarının bilgilerini etkili olarak kodlaması için gereklidirler. Günümüzde, Oiled (Sean Bechhofer et al., 2001) ve Protégé 2000(W. Grosso et al., 2000) gibi araçlar yardımıyla DAML+OIL ve OWL

ontolojileri yaratılabilmektedir. Aynı zamanda, FACT (Horrocks, 1998), ve RACER (Haarslev et al., 1999) çıkarsama motorlarına bağlantı yapılarak ontolojilerin tutarlılık kontrolü (*consistency check*) yapılabilmektedir. Bununla birlikte, DAML+OIL ile birlikte kullanılmak üzere ileri bir sorgulama aracının geliştirilmesine de devam edilmektedir. Halihazırda, çoğu sorgulama dilleri (Karvounarakis et al., 2002) RDF(S) üçlüsü üzerinde çalışmaktadır. Bundan dolayı, DAML+OIL dilinin bütün özellikleri sağlanmamış ve özellikle bilgi tabanlarının ileri seviyede sorgulanmasını gerçekleştirecek çıkarsama motorları henüz gerçekleştirilememiştir.

4.5. GML ve DAML+OIL

GML ile mukayese edilecek olursa, DAML+OIL daha ileri seviyede bir dildir. GML, bir modelleme dilidir ve DAML+OIL kadar açıklayıcı ve anlamsal olarak zengin değildir. GML'in kullanıcıların kavram yaratmasına müsaade etmeyen, önceden yaratılmış bir sözlüğü vardır. Ayrıca, GML'de bilgi üzerinde yapılacak herhangi bir işlem, yazılım üzerinde ilave bir katman açılmasını gerektirmektedir. Bu işlemin de bilginin kendi açıklaması üzerinde yapılması zorunluluğu vardır. Bununla birlikte, GML'in standartlaştırılmış olarak W3C tarafından sunulmuş olması ve önceden yaratılmış bir sözlüğünün olması nedenleriyle web sayfalarının coğrafi olarak işaretlenmesinde yaygın olarak kullanılmaktadır. Gerçekte, çok düzgün belirlenmiş kavramların bu kadar açıklayıcı bir dil ile gösterilmesine pek ihtiyaç yoktur. 2004 yılında DAML+OIL dilinde geliştirilen Agent Cities Urban Ontolojisi (Agent Cities, 2004), sadece dilin belirli bazı özelliklerini kullanmaktadır.

4.6. Coğrafi Ontoloji

İnternet üzerinde uzman personel tarafından hazırlanan birçok mekansal verinin mevcut olduğu bilinmektedir. İşte bu coğrafi verileri, anlamsal web üzerinde kullanabilmek için, OWL kullanılarak coğrafik ontolojilerin hazırlanmasına ihtiyaç duyulmaktadır. Tez çalışmasında Protégé ve SWOOP yazılımları kullanarak hazırlanan coğrafi ontoloji, EK-1’de sunulmuştur. Coğrafi ontoloji; geliştirilen araç yazılımları test etmek amacıyla uygun coğrafik referanslar sağlamak ve coğrafi detaylar arasındaki topolojik, yön ve mesafe ilişkilerini ifade etmek için geliştirilmiştir. Örneğin, ülke, şehir ve kıta olarak sınıfları ve topolojik ilişkileri ifade etmek.

4.6.1. SWOOP ve Protégé Yazılımlarının Karşılaştırılması

Bu tez çalışması esnasında ontoloji yaratmak amacıyla, önceleri Protégé 3.2 Beta yazılımı kullanılarak, daha sonra ise tez çalışmasının amacına daha uygun olduğu değerlendirilen SWOOP 2.3 Beta 3 yazılımları kullanılmıştır.

Protégé yazılımı, Stanford Üniversitesi’nde “Stanford Medical Informatics” çalışma grubu tarafından geliştirilmiştir. (<http://protege.stanford.edu>, 2006)

SWOOP yazılımı, Maryland Üniversitesi’nde Prof. James Handler başkanlığındaki bir çalışma grubu tarafından geliştirilen web tabanlı bir ontoloji yaratma ve gösterme aracıdır. Kullanıcı arayüzü olarak temel web tarayıcısına benzer şekilde adres çubukları, menüler, vb. özellikleri içermektedir (<http://www.mindswap.org/2004/SWOOP>, 2006). Protégé yazılımına göre farklılıkları aşağıda sıralanmıştır:

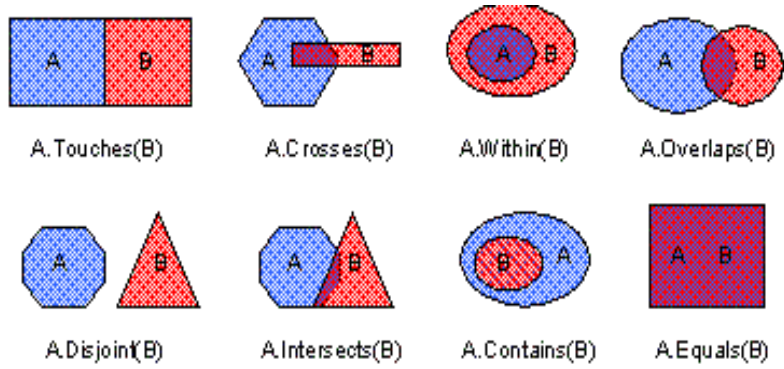
- Ontolojilerin web ortamından yüklenmesi ve birbirleri arasında dolaşımı oldukça basittir.

- Birden fazla ontoloji aynı anda yüklenebilmektedir.
- Ontoloji, “class”, “property” ve “individual” özelliklerinin yüksek seviyede ve ulaşılabilir durumda kolayca gösterimi (render) mümkündür.
- Ontoloji ve varlıklarının çeşitli sözdizimlerde gösterimi (RDF/XML, the OWL Abstract Syntax, Turtle) mümkündür.
- Ontolojilerdeki açıklamaların (annotations) Annotea Protokolü kullanarak paylaşımı daha kolaydır.
- Sınıf hiyerarşisinin dairelerle gösterimi mümkündür.

4.6.2. Topolojik İlişkiler

OGC’nin Basit Detay Tanımlamalarının topolojik ilişkiler standartları Şekil 4.1’de gösterilen aşağıdaki sekiz ilişkiyi içermektedir (Clemintini et al., 1995; Egenhofer et al., 1990):

- eşittir (equals)
- ayrıktır (disjoints)
- kesişir (intersects)
- temas eder (touches)
- çaprazlar (crosses)
- içinde (within)
- içeririr (contains)
- örtüşür (overlaps)



Şekil 4.1: Topolojik İlişkiler

RDF olarak yaratılan tipik topolojik ilişkiler, Şekil 4.2’de gösterilmiştir.

```
<rdf:RDF
  xmlns:j.0="http://www.mindswap.org/2003/owl/geo/geoRelations.owl#"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:j.1="http://www.mindswap.org/2003/owl/geo/geoFeatures.owl#" >
  <rdf:Description rdf:nodeID="A0">
    <rdf:type rdf:resource="http://www.mindswap.org/2003/owl/geo/geoFeatures.owl#Polygon"/>
    <j.1:xyCoordinates>138.75,75.5 ...</j.1:xyCoordinates>
  </rdf:Description>
  <rdf:Description rdf:about="http://nurl.org/feature/10">
    <j.0:intersects rdf:resource="http://nurl.org/feature/11"/>
    <j.0:crosses rdf:resource="http://nurl.org/feature/11"/>
    <j.1:shape rdf:nodeID="A0"/>
    <j.0:overlaps rdf:resource="http://nurl.org/feature/11"/>
  </rdf:Description>
  <rdf:Description rdf:nodeID="A1">
    <j.1:xyCoordinates>154.25,90.5 ...</j.1:xyCoordinates>
    <rdf:type rdf:resource="http://www.mindswap.org/2003/owl/geo/geoFeatures.owl#Polygon"/>
  </rdf:Description>
  <rdf:Description rdf:about="http://nurl.org/feature/11">
    <j.0:intersects rdf:resource="http://nurl.org/feature/10"/>
    <j.1:shape rdf:nodeID="A1"/>
    <j.0:overlaps rdf:resource="http://nurl.org/feature/10"/>
    <j.0:crosses rdf:resource="http://nurl.org/feature/10"/>
  </rdf:Description>
</rdf:RDF>
```

Şekil 4.2: Topolojik İlişkiler Ontolojisi
(<http://www.mindswap.org>, 2006)

Ontoloji, bir temel nesne, referans nesnesi ve düşünce çerçevesi (frame of reference) ile birlikte uygulanacak kantitatif (nicel) ve kalitatif (nitelikli) mesafe ilişkilerine imkan vermektedir (Kuhn, 2003). Örneğin, yönlerin kalitatif (nitelikli) tanımı olarak, esas (cardinal) yönler 8 model uygulanabilir: Kuzey, Kuzeydoğu, Doğu, Güneydoğu, Güney, Güneybatı, Batı, Kuzeybatı.

Ayrıca, bu yön ilişkileri iki kısıtlama (*constraint*) seviyesinde tanımlanabilmektedir:

A isNorthOf B to be true if the northern most point of A is further north than northern most point of B

A is CompletelyNorthOf B to be true if the southern most point of A is north of the northern most point of B

Therefore, if A isCompletelyNorthOf B, the relation A isNorthOf B is also true.

Coğrafi detay sınıfları, RDF temellerine dayalı şekilde, diğer ontoloji ve isimuzayları kullanarak *label*, *administrative codes*, *statistical data* gibi özelliklere sahip olabilirler. Bu, kullanıcılara mekansal verilerle, anlamsal web üzerinde açıklanan herhangi formdaki bilgiler (*spatial* ve *non-spatial* veriler) ile *reasoning* ve *inference*'ı hızlandırmaya yarayan OWL (*sameAs* predicate gibi) ile ilişkilendirmemizi imkan vermektedir.

Coğrafi örnekler, ESRI ArcExplorer'ı kullanılarak mekânsal veri kümeleri içerisinde GML'e çevrilebilmektedir. Daha sonra, XSLT style sheet kullanılarak OWL'a çevrilebilir. Özel bir URI'e tahsis ederek, bu bilgi ağ üzerinde paylaşıma açılabilir. Şekil 4.3'de örnek bir olgu gösterilmiştir.

```
<geo:Country rdf:ID="Türkiye">
  <rdfs:label>Türkiye</rdfs:label>
  <geo:shape>
    <geo:MultiPolygon rdf:nodeID="TurkiyeShape">
      <geo:xyCoordinates>
        -171.848052 9.218889, ...
```

```

</geo:xyCoordinates>
<geo:xyCoordinates>
.....
</geo:xyCoordinates>
</geo:MultiPolygon>
</geo:shape>
</geo:Country>

```

Şekil 4.3: Örnek bir olgu (isimuzayı göz ardı edilmiştir.)

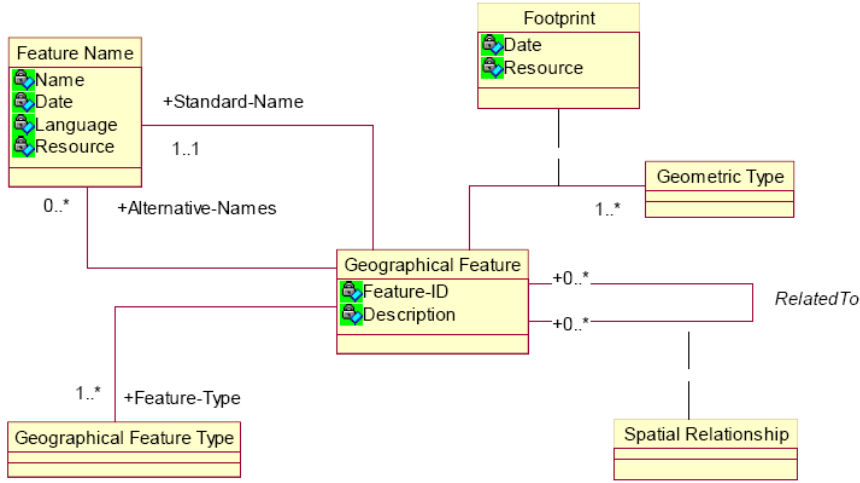
Benzer konuda yapılan çalışmalara örnek olarak, İngiltere Cardiff Üniversitesi'nde yürütülmekte olan SPIRIT (Spatially Aware Information Retrieval on the Internet), (Jones et al., 2001, 2003) projesi gösterilebilir. Bu proje, Avrupa Anlamsal Web projesinin bir parçası olarak faaliyet göstermekte ve web tabanlı, akıllı bir coğrafik veri arama motoru sunmayı amaçlamaktadır. SPIRIT sisteminin merkezinde, sistemin farklı parçalarına destek veren bir coğrafik ontoloji bulunmaktadır. Ontoloji,

- sorgunun belirsizliğinin giderilmesinde (disambiguation),
- istenen sorgunun sorgu terimlerinin genişletilmesinde,
- alınan sorgu sonuçlarının konularına göre sıralanmasında,
- aramanın desteklenmesi için mekansal indekslerin yaratılmasında

ve

- web dokümanları, coğrafi veri kümeleri gibi web kaynaklarının açıklayıcı olarak notlandırılmasında önemli rol oynamaktadır.

SPIRIT projesinde sunulan coğrafi ontoloji Şekil 4.4'de görülmektedir.



Şekil 4.4: Coğrafi Detay Ontolojisinin Ana Şeması
(C.B. Jones et al. 2003)

Coğrafi ontolojinin gösterimi için, bazı gösterim dilleri mevcuttur. Bunlardan bazıları, betim mantığı tabanlı (description logic-based) DAML+OIL (Ian Horrocks, 2002) ve XML tabanlı (OGC Recommendation Paper, 2001)'dir. SPIRIT Ontolojisi, dillerin değerlendirmesinin yapılabilmesi için örnek bir çalışma olarak alınabilir ve her iki dilin örnekleri sunulan fikirleri destekleme amacıyla verilebilir.

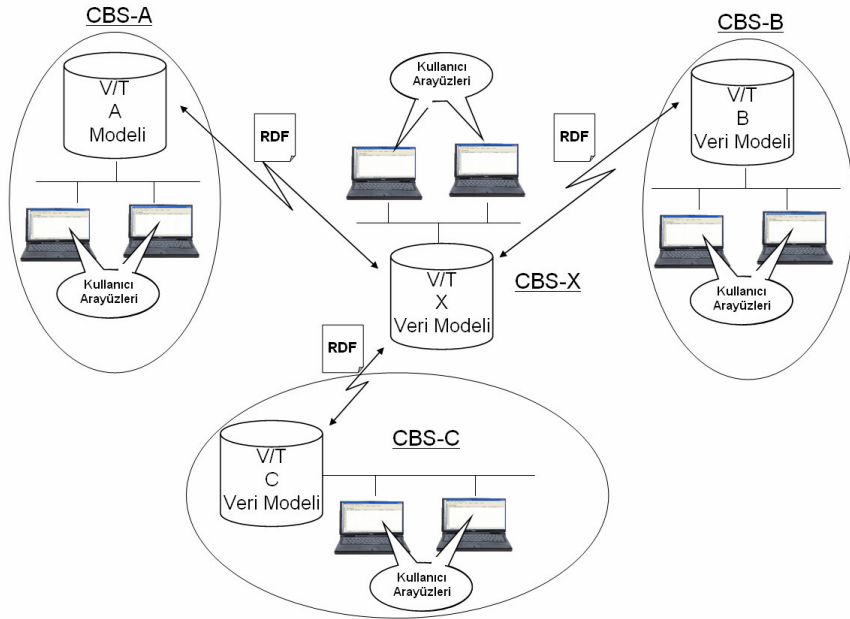
SPIRIT projesinde, hem bir coğrafi ontoloji geliştirme alanıyla ilgili çalışmalar, hem de GML ve DAML+OIL ile ilgili bilgiler sunulmaktadır. Her iki dilin coğrafi ontolojiyi destekleme ve gösterme imkanları ile kısıtlamaları üzerinde çalışılmıştır. Aynı zamanda, bir coğrafi ontoloji için gereken detaylar (features) kümesi önerilmiştir.

DAML+OIL ve diğer ontoloji dillerinin yeteneklerinin tam olarak anlaşılabilmesi için; bir gereksinim tanımlaması yapılması, çeşitli prototip ontolojiler geliştirilmesi, test edilmesi ve her dilin gereksinimi karşılamadaki yeteneklerinin ölçülmesi gereklidir. (Horrocks et al., 2004)

5. UYGULAMA

Bu tez çalışmasında, ayrıntıları üçüncü ve dördüncü bölümlerde açıklanan coğrafi anlamsal web teknolojisi kullanılarak, coğrafi referansların uygulamalar arasında paylaşımı üzerinde çalışılmıştır. Çalışma, genel olarak iki bölümden oluşmaktadır:

- Ontoloji ve olgularının yaratılması,
- Ontolojiyi kullanarak heterojen veritabanlarına sahip uygulamalar arasında coğrafi olarak referans edilmiş verilerin paylaşımı. Örneğin, Şekil 5.1’de üç değişik heterojen CBS üzerindeki belirli coğrafi detayların (yerleşim yeri, hava, deniz limanları, vb.) tek bir CBS uygulamasında bütünleştirilmesi gösterilmiştir.



Şekil 5.1: Heterojen uygulamaların birlikte çalışabilirliği kapsamında coğrafi verilerin paylaşımı

Bahsedilen iki çalışmanın, hem dezavantajları hem de avantajlarını bulma şansı elde edilmiştir. Bu da çalışmaların gelecekteki yönünün belirlenmesini sağlamıştır. Coğrafi ontoloji, tipik coğrafi detay sınıflarını içermektedir. Ayrıntıları dördüncü bölümde açıklanan OWL dili kullanılarak Protégé ve SWOOP editörleri yardımıyla hazırlanmıştır. Dolayısıyla, RDF'e dayalı olan diğer coğrafik ontolojilerle beraber kullanılabilir.

5.1 Ontoloji ve Olgularının Geliştirilmesi

5.1.1. Bilgi Değişim Veri Modeli Ontolojisi

Heterojen sistemler arasında tespit edilen minimum bilgi değişim ihtiyaçlarının oluşturduğu veritabanının (<http://www.mip-site.org>, 2006) bir kısmını içeren ve tüm sistemlerin verilerini ifade etmek için geliştirilen ontolojidir (bdvm.ont). Bu ontoloji yardımıyla tüm modüller sistemdeki nesne türlerini anlayabilmektedirler.

5.1.2. Sorgu ontolojisi

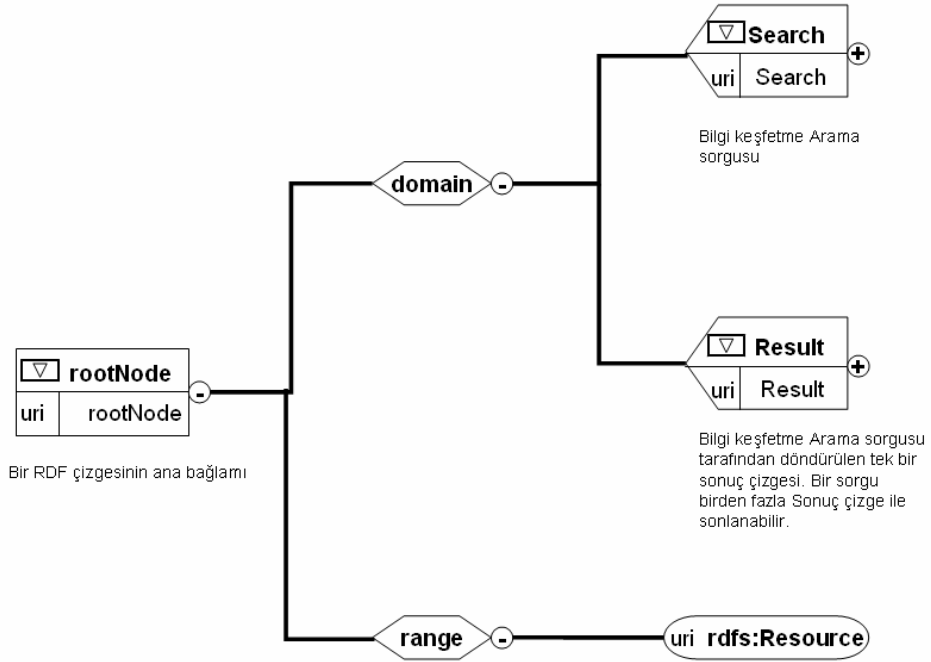
Sorgu oluşturulurken kullanılan ontolojidir. Burada, terimlerdeki farklılıkları göstermek ve anlam birlikteliği oluşturmak amacıyla *sorgu* (query) ve *arama* (search) terimlerini ifade etmekte fayda vardır. *Sorgu*, sadece bilgi çekimini (*information retrieval*) ifade etmek için, *arama* ise, Google arama motorunun ürettiğine benzer bir şekilde bilgi keşfetmek (*information discovery*) amacıyla kullanılmaktadır.

Aslında, kavram olarak ifade etmek istenirse, her iki tipteki istekleri *sorgular* (*queries*-küçük “q” ile) şeklinde belirtmek uygun olur. Burada, *sorgu* ile *arama* arasındaki temel fark, sorgu ontolojisi yardımıyla ortaya konulmaya çalışılmıştır. Sorgu ontolojisi, halen kullanılmakta olan mevcut araç yazılımlar ve bileşenler için, geriye dönük uyumluluğu sağlamak amacıyla, hem OWL-Full, hem de RDF Şema cinsinde geliştirilen üst-veri tanımlamalarını içermektedir. Jena

kullanılarak bu ontolojiler yüklenebilir ve java yazılımları için kullanılabilirler. Sorgu ontolojisi 3 önerme (*predicate*) tanımlamaktadır: *rootNode*, *fuzzyMatchFactor* ve *maxResults*

rootNode :

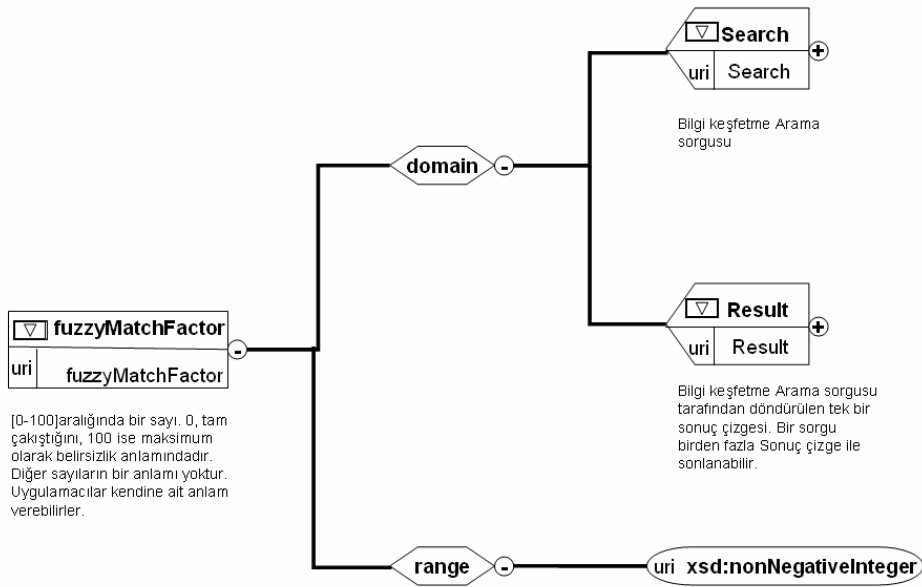
Şekil 5.2’de grafiksel üst-veri olarak ifade edilen *rootNode*, zorunlu bir önermedir. Eğer, bir *arama* içerisinde yoksa, alıcı taraf *aramayı* yerine getirmez ve hata mesajı alır.



Şekil 5.2: *rootNode* Üst-Verisi

FuzzyMatchFactor:

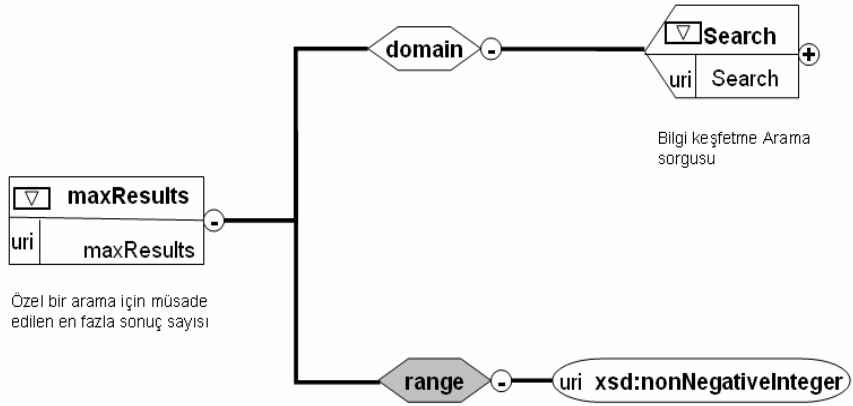
Şekil 5.3’de grafiksel üst-veri olarak ifade edilen *FuzzyMatchFactor*, zorunlu olmayan bir önermedir. Eğer, bir *arama* içerisinde mevcut değilse, üreten taraftaki *aramanın FuzzyMatchFactor* değeri 100’müş gibi döndürülür.



Şekil 5.3: *FuzzyMatchFactor* Üst-Verisi

maxResults:

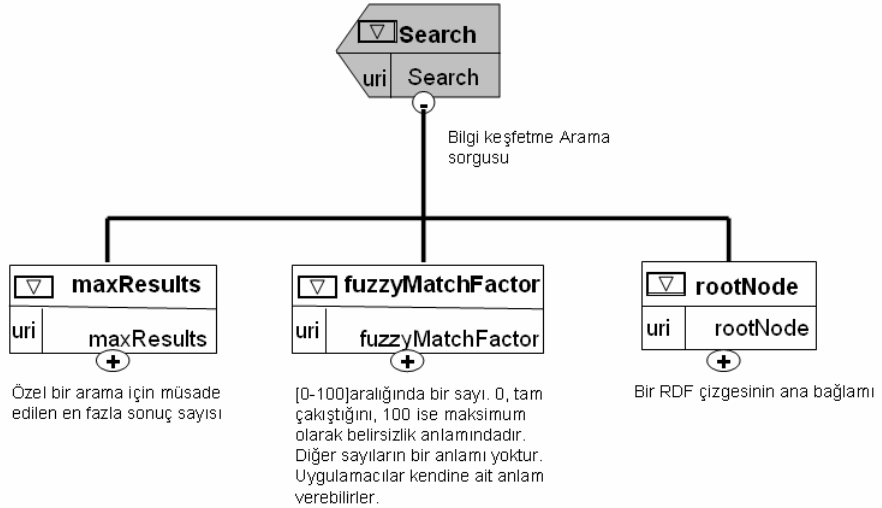
Şekil 5.4’de grafiksel üst-veri olarak ifade edilen *maxResults*, zorunlu olmayan bir önermedir. Eğer, bir *arama* içerisinde mevcut değil ise, üreten taraftaki *arama*, gelişigüzel bir arama sonucu miktarı döndürür.

Şekil 5.4: *maxResults* Üst-Verisi

Sorgu ontolojisi 2 sınıf tanımlamaktadır: *Search* ve *Result*

Search:

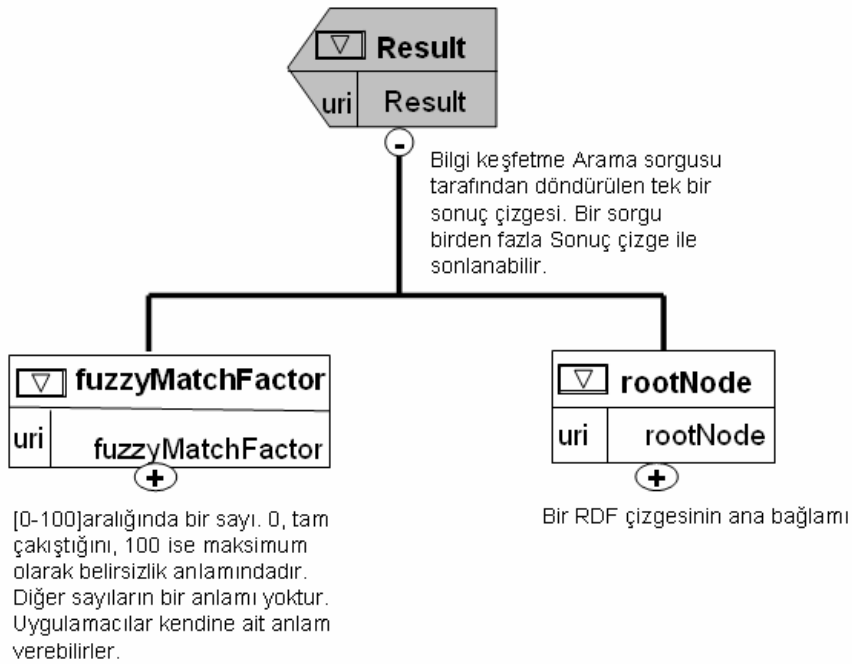
Her sorgu *araması*, bu sınıfın bir örneği veya bireyi (instance/individual) olmak zorundadır.

Şekil 5.5: *Search* Üst-Verisi

Yukarıda açıklanan üç önermeden farklı olarak, *Search*, arama yapmaktan sorumlu varlığı (*entity*) belirlemek amacıyla *dc:creator* ile açıklanır. *Search* sınıfı, Şekil 5.5’de grafiksel üst-veri olarak ifade edilmiştir.

Result:

Her sonuç döndürüldüğünde, bir *Result* olgusu içermesi gerekmektedir. Daha sonra, bu olgu, belirli bir üst-veri ile ilişkilendirilir. *Result* sınıfı, Şekil 5.6’da grafiksel üst-veri olarak ifade edilmiştir.



Şekil 5.6: Result Üst-Verisi

Örnek olarak bir arama yapmak amacıyla, “Ahmetli” ismi olan bütün ilçeler istenirse:

```

APPL.Ilce.ilce = new APPL.Ilce();
ilce[APPL.ObjectItemName] += Util.CreateTypedLiteral("string",
"Ahmetli");
Search search = new Search();
search.rootNode += ilce;
search.fuzzyMatchFactor += Util.CreateTypedLiteral("integer", "0");
search.maxResults += Util.CreateTypedLiteral("integer", "10");
search[DC.creator] += Util.CreateTypedLiteral("string",
MainForm.RegistrationName);

```

RDF/XML (SemWeb RdfXMLWriter class)

```

<rdf:RDF xmlns:dc="http://purl.org/dc/elements/1.1/"
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:xsd="http://www.w3.org/2001/XMLSchema#">
  <bdvm:Ilce rdf:nodeID="genid2">
    <bdvm:ObjectItemName
rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
Ahmetli
    </bdvm:ObjectItemName>
  </bdvm:Ilce>

  <semweb:Search rdf:nodeID="genid1">
    <dc:creator
rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
Arayıcı - mkomesli
    </dc:creator>
    <semweb:rootNode rdf:nodeID="genid2" />
    <semweb:fuzzyMatchFactor
rdf:datatype="http://www.w3.org/2001/XMLSchema#integer">
0
    </semweb:fuzzyMatchFactor>
    <semweb:maxResults
rdf:datatype="http://www.w3.org/2001/XMLSchema#integer">
10
    </semweb:maxResults>
  </semweb:Search>
</rdf:RDF>

```

5.1.3. Coğrafi Ontoloji

Projede kullanılan coğrafi mekansal yer varlıkları (*geospatial entity*)'nin ifade edilebilmesi amacıyla geliştirilen, ortak bir sözlüktür. WGS84 datumuna göre yer yüzeyi ve coğrafi mekansal konseptlerin ifade edilebilmesi için imkan sunmaktadır. Proje kapsamında geliştirilen ontoloji (GeoOnt.owl), EK'te sunulmuştur.

5.2 Coğrafi Verilerin Uygulamalar Arası Paylaşımı

Daha önceden geliştirilmiş ve halen kullanılan sistemlerden bilgiyi keşfetmek, çekmek ve yararlı hale getirmek için, bir tür protokol yerine geçebilecek uygulama geliştirilmiştir. Mevcut teknolojiler yeni bir şekilde birleştirilerek, bu amaca ulaşılmaya çalışılmıştır. Bunun için, uygulama küçük ve basit tutulmuştur. Bu yaklaşım ile, uygulama maliyeti azalmış ve eski sistemler tarafından benimsenme olasılığı arttırılmıştır.

Uygulamanın ana fonksiyonları, bir SOAP web servisi olarak geliştirilmiştir. Bu web servisi, diğerlerinden farklı olarak, yapısal veri içeren diğer uygulamalar ve sunumcular tarafından başarılı bir şekilde gerçekleştirilmeye imkan vermektedir. Bu şekilde kolay uygulanabilir olması, uygulamanın, geleneksel sistemlere olan bağımlılık engelini ortadan kaldıracağı ve kolayca benimsenmesine imkan vereceği değerlendirilmektedir.

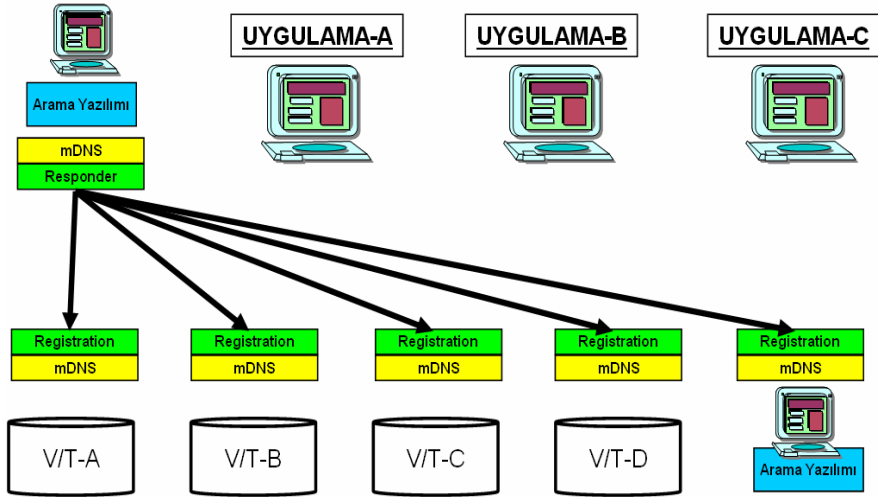
İstemci tarafında, mevcut bilgi kaynaklarını keşfederek ortaya çıkartacak bir yetenek mümkündür. Kullanılmakta olan eski sistemler, halen belirlenmiş amaca yönelik olarak veri sunmakta olduklarından dolayı, bu yeni yaklaşımda önemli bir rol oynamaktadırlar. Ortaya çıkan yeni uygulamalar ise, verinin kaynağı olan bu eski sistemler üzerine inşa edilmektedir. Bu proje kapsamında gerçekleştirilen uygulama, anlamsal web'e dayalı bu yeni yaklaşımları sunmaktadır.

Uygulama içerisinde kullanılan temel fonksiyonlar şunlardır:

- HTTP
- Zeroconf (Multicast DNS/Bonjour) ([http://www. zeroconf.org](http://www.zeroconf.org), 2006); ([http://www.multicastdns. org](http://www.multicastdns.org) , 2006)
- Simple Object Access Protocol (SOAP)

- Resource Description Framework (RDF/XML) (<http://www.w3.org/XML>, 2006); (<http://www.w3.org/RDF> , 2006)

Uygulamadan amaç; kısaca diğer uygulamaların bilgi isteklerini karşılamaktır. Uygulama, dinamik kayıt olma ve protokol sağlayıcıların keşfedilmesi işlevlerini yerine getirmektedir. Bu iki yeteneğin birleşmesi ile, bir bilgisayar ağına irtibatlanmış istemcinin hızlı bir şekilde bütün bilgi sağlayıcıları bulmasına ve onları ilgi alanına göre sorgulamasına imkan vermektedir. (Şekil 5.7)



Şekil 5.7: Uygulamanın gösterimi

Sorgu sonucunu kullanışlı bir şekle getirebilmek için, elde edilecek bilgiye bazı yapılar uygulanmıştır. Bilgi keşif protokolü, sorgu sonucu dönen bilgiyi RDF şeklinde getirmektedir. RDF, üçüncü bölümde ayrıntıları açıklandığı üzere, varlıkları (*entitiy*), özniteliklerini (*attributes*) ve bunlar arasındaki ilişkilerini tanımlamak maksadıyla yönlendirilmiş çizge (*directed graph*) kullanmaktadır. Buradan anlaşılabacağı üzere, RDF, gerçek dünya üzerindeki nesneleri tanımlamak ve birbirleri ile

ilişkilendirebilmek için ideal bir çalışma ortamıdır. Böylelikle, sistemler arası keşfedilerek ve çekilerek değişecek bilgiler, bilinen ve tekrar kullanılabilen bir yapıda olacaktır.

Servis Keşfetme Protokolü (*Service Discovery Protocol*) sayesinde SOAP web servisleri, ortama duyurulmakta, ve böylelikle uygulamalara uyumlu hale gelmektedir. Bu amaçla, bir uygulamanın ortama kayıt olma ve keşfedilmesi için “Zeroconf” bir ağda “MulticastDNS (mDNS)” kullanılmaktadır. (<http://www.multicastdns.org>, 2006)

mDNS kayıtları, o anda uygun olan her servis için gerekli temel bilgileri içermektedir. Bu temel bilgiler şunlardır:

- Bilgi sağlayıcı sistemin IP adresi,
- Bilgi sağlayıcı sistemin dinlediği port,
- Bir SOAP isteğinin çağrılabilmesi için gerekli “path” (Bir URL bağlantısı da olabilir. Örneğin, <http://host/path> veya [/path](http://host/path)).
- Uyumlu olmayan sistemleri ayıklayabilmek için kullanılan uygulamanın sürüm numarası.

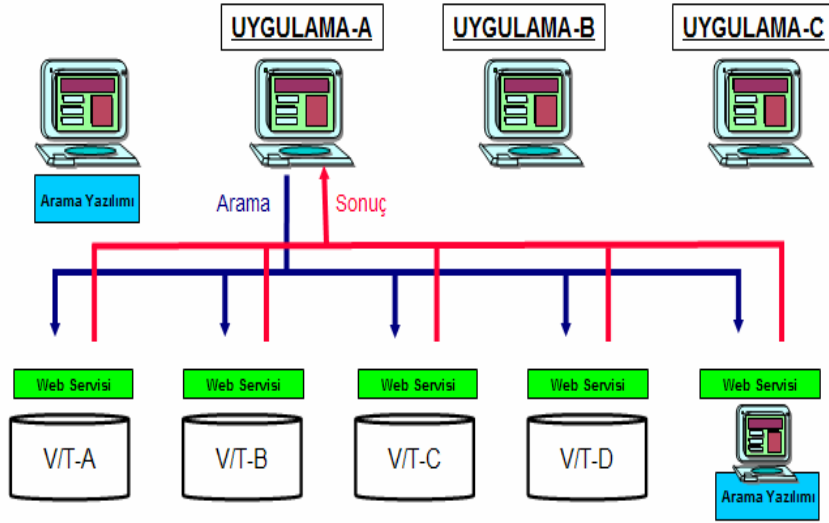
Burada, mDNS kayıtlarının, servis tarafından yayımlanan metodların isimlerini içermesine gerek duyulmamıştır.

5.2.1 Aramanın Yapılması

Bir web servisi keşfedildiği zaman, RDF formatında (N-triple) bir Bilgi Keşif İsteği kodlanır. Daha sonra, sorgunun yapılması için aşağıdaki bileşenler, bir SOAP isteği olarak yapılandırılır (Şekil 5.8). SOAP isteği, HTTP protokolü kullanılarak yerine getirilir:

- mDNS’den:
 - servis sağlayıcının IP adresi
 - servis sağlayıcının *Port*’u

- servise olan HTTP *path*'i
- İstemci uygulamasından:
 - Sorguyu tanımlayan RDF/XML dokümanı
- Sabitler (*constants*)
 - Çağrılacak SOAP metodu (*Search()* metodu)



Şekil 5.8: Aramanın yapılması

SOAP isteğinin çağırılması üzerine, servis sağlayıcı tarafından, sonuçların tanımlandığı RDF çizgesi içeren bir XML dokümanı döndürülmektedir. Hemen ardından, ayrıntıları 5.2.2. bölümünde açıklandığı şekilde SOAP cevabı hazırlanarak kodlanır. Daha sonra, Servis Keşfetme Protokolünün uygulayan servisler bulunur. mDNS’de Sorgu protokolü “_tinf-query._tcp.local.” tipi ile gösterilmektedir. İlgili detaylar kullanılarak, SOAP sorgusu için bir URI yaratılır. Örneğin, <http://172.31.85.88:80/BICCRdfWebService/Service.asmx>.

Örnek olarak, arazi üzerinde koordinatları bilinen göller bulunmak istenirse, bu aramanın RDF cinsinden kodlanması aşağıdaki gibi olmaktadır:

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  <bdvm:Gol rdf:about="?gol">
</bdvm:Gol>
  <semweb:Search rdf:nodeID="search">
    <semweb:rootNode rdf:resource="?gol" />
  </semweb:Search>
</rdf:RDF>
```

Sorgunun çalıştırılması için, istek, 5.2.2. bölümünde açıklandığı şekilde SOAP cevabı hazırlanarak kodlanır. Daha sonra, istek, HTTP POST kullanılarak sunucuya gönderilir.

```
POST /RdfWebService/Service.asmx HTTP/1.1
Host: 172.31.85.88
Content-Type: text/xml;
Content-Length: 720
SOAPAction: http://173.30.85.88/RdfWebService/Service.asmx/Search
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <Search>
      <rdf:RDF
        xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
        <bdvm:Gol rdf:about="?gol">
</bdvm:Gol>
        <semweb:Search rdf:nodeID="search">
          <semweb:rootNode rdf:resource="?gol" />
        </semweb:Search>
      </rdf:RDF>
    </Search>
  </soap:Body>
</soap:Envelope>
```

Bir RDF grafiği formatındaki sorgu sonuçları, yine 5.2.2. bölümünde açıklandığı şekilde kodlanır ve HTTP cevabı olarak döndürülür. Sistem tarafından döndürülen cevap aşağıda görülmektedir.

```
HTTP/1.1 200 OK
Server: Microsoft-IIS/5.1
Date: Fri, 17 Jun 2006 14:29:02 GMT
X-Powered-By: ASP.NET
X-AspNet-Version: 1.1.4322
Cache-Control: no-cache
Pragma: no-cache
Expires: -1
Content-Type: text/xml; charset=utf-8
```

```

Content-Length: 1177
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <Search>
      <rdf:RDF xmlns:ns2="http://ege.edu.tr/develop/location#"
        xmlns:ns1="http://ege.edu.tr/develop/bdvm#"
        xmlns:ns4="http://ege.edu.tr/develop/symbology#"
        xmlns:ns3="http://ege.edu.tr/geeont#"
        xmlns:dc="http://purl.org/dc/elements/1.1/"
        xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
        <ns1:Gol
          rdf:about="http://192.168.1.108/ORBAT/DetailTrans.aspx?ORBAT=0ac86800-9674-
          11d7-8c94-00065b57d4a8xxxGOL">
            <ns1:featureNicknameName>
              Karasu
            </ns1:featureNicknameName>
            <ns1:objectItemName>Karasu</ns1:objectItemName>
            <ns1:golId>0ac86800-9674-11d7-8c94-00065b57d4a8</ns1:golId>
            <ns1:golFormalAbbreviatedName>
              Karasu
            </ns1:golFormalAbbreviatedName>
            <ns1:objectItemAlternateIdentificationText>
              Karasu
            </ns1:objectItemAlternateIdentificationText>
            <ns1:organisationId>
              0ac86800-9674-11d7-8c94-00065b57d4a8
            </ns1:organisationId>
            <ns2:locatedAt rdf:nodeID="genid2" />
            <ns3:Speciality>balikcilik</ns3:Speciality>
            <ns3:Type>acisu</ns3:Type>
            <dc:title>Karasu</dc:title>
            <ns3:Country>Turkiye</ns3:Country>
            <dc:publisher>egeont</dc:publisher>
            <ns3:Hostility>Unknown</ns3:Hostility>
            <ns4:affiliation>Unknown</ns4:affiliation>
            <ns4:visualization>SUGP-----*****</ns4:visualization>
          </ns1:Gol>
          <ns2:Point rdf:nodeID="genid2">
            <ns2:coordinates rdf:nodeID="genid1" />
          </ns2:Point>
          <ns2:GeoRef rdf:nodeID="genid1">
            <ns2:altitude>0</ns2:altitude>
            <ns2:latitude>17.2033888888889</ns2:latitude>
            <ns2:longitude>17.2033888888889</ns2:longitude>
          </ns2:GeoRef>
        </rdf:RDF>
      </Search>
    </soap:Body> </soap:Envelope>

```

Cevaplayıcı, temel *mDNS* desteği sağlayan bir *Windows* servsidir. Aynı zamanda, *Apple*'ın *Bonjour* yazılımı üzerinde C# dilinde geliştirilen bir paketleyici (*Wrapper*)'dır. (<http://developer.apple.com/darwin/tools/cvs/howto.html>, 2006). Kısaca, “*Cevaplayıcı*, *Arayıcı* yazılımının yerel ağdaki sunumcuları bulabilmesi için çalışması gereken *mDNS* tabanlı MS *Windows* servsidir” denilebilir.

Kayıt edici, kullanıcı tarafından ayarlanabilen ve tek bir mDNS servisini otomatik olarak ortama sunan bir bileşendir. *JmDNS* ise Java programlama dilinde yazılmış bir mDNS kütüphanesinin yanısıra *Kayıt Edici* ve *Keşfedici Servisler* sunmaktadır. (<http://jmdns.sourceforge.net>, 2006). Kısaca, “JmDNS paketleyici (*Wrapper*), Java programlama dili kullanılarak hazırlanan web servisini lokal ağda bulunan istemcilere görünür hale getiren yazılımdır” denilebilir.

5.2.2 SOAP Kodlandırması

Bu bölümde, proje kapsamında kullanılan SOAP protokolü yardımıyla dokümanların nasıl basitçe kodlandığı ve kodlarının çözüldüğü anlatılmaktadır. Sabit bir kodlama ve kod çözme işlemi, heterojen olmayan sistemlerin birlikte çalışabilirliğini sağlanması açısından önem taşımaktadır. (<http://www.w3.org/Protocols>, 2006) ve (<http://www.w3.org/2000/xp/Group>, 2006).

SOAP protokolü web servislerindeki metotlar için kullanılmaktadır. SOAP kodlaması, ilgili dokümana harfi harfine riayet edilmesi gereken bir yaklaşım gerektirir. Temel olarak bilinen bu yaklaşıma rağmen, uygulamalar pratikte değişiklik gösterebilmektedir. Aşağıdaki örnekte dokümanların kodlanması ve kod çözülmesi işlemlerinde SOAP’ın kullanılışı gösterilmektedir.

```
POST /Path/To/Service.asmx HTTP/1.1
Host: 127.0.0.1
Content-Type: text/xml; charset=utf-8
Content-Length: 720
SOAPAction: http://hostname/Path/To/Service.asmx/SoapMethod

<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <SoapMethod>
      <Document />
    </SoapMethod>
  </soap:Body>
</soap:Envelope>
```

SoapMethod, web servis sağlayıcısı tarafından tanımlanmıştır ve yukarıdaki örneğe uygun olarak kodlanmıştır. *Document*, servis sağlayıcısı tarafından istenen bir içeriktir ve *method* ile ilgili argümanları tedarik eder. HTTP başlığı, SOAP isteği için uygun *SOAPAction* başlığı içerir. *SOAPAction* başlığı, yukarıdaki örnekte görüldüğü gibi *SoapMethod*'unu tipik bir URI'ye eklemekten ibarettir.

Sistemde bir servis, ortama her zaman için *Search* adı verilen bir *SoapMethod*'unu oluşturmaktadır. Aşağıda ortama gönderilecek örnek bir SOAP mesajı olarak, *Search* bilgi keşif isteği görülmektedir.

```
POST / HTTP/1.1
CONTENT-LENGTH: 761
EXPECT: 100-continue
SOAPACTION: http://172.31.85.45:9475//Search
CONTENT-TYPE: text/xml
HOST: 172.31.85.45:9475
CONNECTION: Keep-Alive
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <Search>
      <rdf:RDF ...>
        [ ... query RDF/XML tree ... ]
      </rdf:RDF>
    </Search>
  </soap:Body>
</soap:Envelope>
```

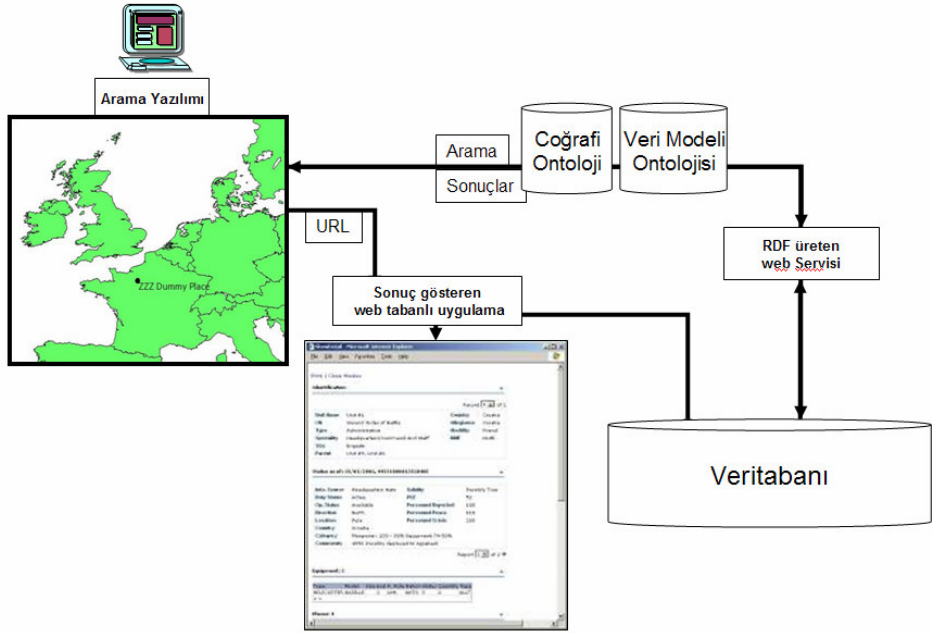
Bilgi sağlayıcı tarafında ise, *SoapMethodResponse* tanımlanır. Genellikle, Cevap (*Response*) eklenmiş bir *SoapMethod*'udur. Bundan dolayı *Query* isimli metod *QueryResponse* olarak adlandırılmıştır. İstenilen şey olarak, bir RDF/XML dokümanı olan *Document* döndürülür.

```
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <SoapMethodResponse>
      <Document />
    </SoapMethodResponse>
  </soap:Body>
</soap:Envelope>
```

Çalışma kapsamında *Search()* metoduna cevap, daima *SearchResponse* içerisinde paketlenmiş (wrapped) olacaktır. *Search* bilgi keşif isteğine cevap olarak alınan SOAP mesajı aşağıdadır:

```
SoapHelper: Content-Type: text/xml
SoapHelper: SOAPAction: http://173.35.85.45:9475//Search
SoapHelper: Content-Length: 761
SoapHelper: Expect: 100-continue
SoapHelper: Connection: Keep-Alive
SoapHelper: Host: 172.31.85.45:9475
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <SearchResponse>
      <rdf:RDF>
        [ ... result RDF/XML trees ... ]
      </rdf:RDF>
    </SearchResponse>
  </soap:Body>
</soap:Envelope>
```

Sonuç olarak, Şekil 5.9’da görüldüğü gibi, sorgu sunucu dönen veriler, temel nesneleri içeren veri modeli (BDVM) ontolojisi kullanılarak bir RDF dosyasına (localdata.rdf) yazılmaktadır. Dolayısıyla, aynı veri modelini kullanmasa dahi uygulamalar arasında ontoloji kullanılarak veri paylaşımı gerçekleştirilmektedir. Burada bir CBS veritabanından yerleşim merkezi ile ilgili bilgilere, koordinat verisinin de mevcut olması halinde sayısal harita üzerinde gösterilebilmekte ve diğer uygulamaların istifadesine web tabanlı olarak sunulabilmektedir.



Şekil 5.9: Bilgi paylaşımı

Bu tez çalışmasında, yazılımlar genelde C# kullanılarak geliştirilmiştir. Ancak, Java’da bir örnek yazılım geliştirilmiştir. Geliştirilen yazılımlar aşağıda açıklanmaktadır:

Arama Yazılımı: Geliştirilen sunumcuyu değerlendirebilmek için C# programlama dili kullanılarak geliştirilmiş bir istemcidir. Sadece XP işletim sistemi kurulu makinalarda denenmiştir. .NET (runtime environment--dotnetfx.exe) kurulumu gerektirmektedir. Sorgulama yapıldığında, kendine ait bir RDF veri tabanından bilgi okumaktadır. Ancak kurulumdaki bir sorundan dolayı bu RDF dosyasını (*LocalData.rdf*) *Arama yazılımı* bir kere çalıştırıldıktan sonra oluşan “C:\DocumentsandSettings\CurrentUser\LocalSettings\ApplicationData\Arama\CurrentUser” klasörüne kopyalanmalıdır. Örnek olarak, içerisinde 2 şehir ve 6 göl bilgisi bulunmaktadır. Ayrıca, tarayıcı penceresi

içerisinde farenin sağ tuşuna basarak yenilenmesi (refresh) gerekmektedir.

.NET Runtime (dotnetfx.exe): Arama yazılımını kullanmak için gerekli olan .NET runtime env. kurulum dosyasıdır. (<http://www.microsoft.com/downloads/details.aspx?FamilyId=262D25E3-F589-4842-8157-034D1E7CF3A3&displaylang=en>, 2006)

Service.asmx.cs: Sunumcusunun ana metodu olan “search”ın gerçekleştirimi için örnek dosyadır. C# programlama dili kullanılarak yazılmıştır. Java programlama dili kullanılarak hazırlanacak bir Web Servisinde de temel olarak “*public XmlNode Search(...)*” metodunun hazırlanması gerekmektedir. RDF dosyalarının işlenmesi için de C# programlama dili kullanılmıştır. Java programlama dili için aynı işlemleri yapan Jena ürünleri mevcuttur. (<http://prdownloads.sourceforge.net/jena/Jena-2.3.zip?download>, 2006) adresinden Java’da RDF eşleştirmesi için Jena 2.3’e erişilebilmektedir.

Bu tez çalışmasında hedeflenen; bir CBS uygulama yazılımının veritabanından, coğrafi veriler (lokasyon) ile ilgili bilgileri anlamsal web teknolojisi kullanarak alınmasıdır. Bu hedefi gerçekleştirmek için, C# programlama dili kullanılarak *Arama Yazılımının* geliştirilmiştir. Uygulamanın işleyişi içerisinde, veriyi sağlayan sistemin kullanıcı sisteme sorgu sonucu gönderdiği veriden daha detaylı bilgiye erişmek isteyen kullanıcılar için Web sayfaları sağlanması kolaylığı da mevcuttur.

Temel olarak bu veri alışverişi sistemlerin gerçek kapasitelerini yansıtmamaktadır. Ancak, ilk entegrasyon adımı olarak kabul edilebilir ve devamında daha detaylı bilgi alışverişi sağlanabilecektir.

Anlamsal web teknolojisi kullanılarak geliştirilen istemci ve sunumcunun çalışması şu şekilde özetlenebilir:

Yerel bilgisayar ağında bulunan sorgu sunucuları arasından entegre olunacak sunucu seçilir. “Subject Type” olarak coğrafi ontolojide bulunan coğrafi referanslar (GeoRef) seçilir ve sorgu gerçekleştirilir. Sorgu sonucu veritabanında bulunan coğrafi konumlara ait (Geoloc) (yerleşim yeri, havaalanı, göl, vb.) bilgiler listelenir. Her yer bilgisine ait web sayfasının adresi de sorgu sonucu döndürülür. Böylelikle, veriyi sağlayan sisteme daha çok bilgi için bu köprüler üzerinden dönülebilmektedir. Seçilen Coğrafi konuma ait web sayfası görüntülenebilir. Coğrafi konum bilgileri, veri modeli ontolojisindeki “Object Id” alanı ile de sorgulanabilir.

```
<rdf:RDF xmlns:loc="http://ege.edu.tr/Location#"
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
<loc:GeoRef rdf:about="#point">
<loc:latitude>42</loc:latitude>
<loc:longitude>34</loc:longitude>
</loc:GeoRef>
</rdf:RDF>
```

Uygulamalar arasında değiştirilen ve “localdata.rdf” olarak dosyanın çıktısı, RDF olarak aşağıda gösterilmiştir. Bahse konu dosya içerisinde örnek olarak bir şehir ve bir göl bilgisi görüntülenmektedir.

```
<?xml version="1.0" encoding="utf-8" standalone="yes"?>
<rdf:RDF xmlns:dc="http://purl.org/dc/elements/1.1/"
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:ns1="http://ege.edu.tr/develop/bdvm#">
  <ns1:Gol rdf:about="urn:x-semweb:semweb-gol4">
    <ns1:objectItemName
rdf:datatype="http://www.w3.org/2001/XMLSchema#string">DUMMY-
4</ns1:objectItemName>
    <dc:title
rdf:datatype="http://www.w3.org/2001/XMLSchema#string">SEMWEB Arama
Yazilimi Yapay Gol 4</dc:title>
    <dc:publisher
rdf:datatype="http://www.w3.org/2001/XMLSchema#string">SEMWEB Arama
Yazilimi </dc:publisher>
    <dc:description
rdf:datatype="http://www.w3.org/2001/XMLSchema#string">Bu Arama
Yazilimi tarafından ornek amaclı sunulan bir yapay
Goldur.</dc:description>
```

```

</ns1:Gol>
<ns1:Sehir rdf:about="urn:x-semweb:semweb-sehir1">
  <ns1:objectItemName
rdf:datatype="http://www.w3.org/2001/XMLSchema#string">SEHIR-
1</ns1:objectItemName>
    <dc:title
rdf:datatype="http://www.w3.org/2001/XMLSchema#string">SEMWEB
Semweb Yapay Sehir 1</dc:title>
    <dc:publisher
rdf:datatype="http://www.w3.org/2001/XMLSchema#string">SEMWEB
Semweb</dc:publisher>
    <dc:description
rdf:datatype="http://www.w3.org/2001/XMLSchema#string">">Bu Arama
Yazilimi tarafindan ornek amaclı sunulan bir yapay
Sehir dir.</dc:description>
  </ns1:Sehir>

```

6. SONUÇ VE ÖNERİLER

Anlamsal web'in günümüz bilgi teknolojilerine girmesiyle, ontolojiler sadece yapay zeka konularında uğraşanların ilgi odağı olmaktan çıkmıştır. Artık, bilgi paylaşımında bilgisayarların hangi tür bilginin paylaşılacağına karar verilirken ortak bir anlayış sağlamak görevi de ontolojilere verilmiştir.

Bu çalışmada, anlamsal web'e dayalı coğrafi bilgi sistemlerinin coğrafi olarak referans edilmiş bilgileri anlamaları ve birbirleri arasında değişmelerine hizmet etmek amacıyla bir coğrafi ontoloji geliştirilmiştir. Böylelikle, coğrafi veri tanımının ortaya konulması amaçlanmıştır. Web üzerinde coğrafi olarak referans edilen bilgilerin gösterimi ile ilgili olarak ortaya koyulan yaklaşımın, gelecek çalışmalara yol gösterecek şekilde avantaj ve dezavantajları vardır.

Bilgisayarlar, GML ile geliştirilmiş coğrafik referansları, web'deki köprüler gibi esnek olarak işlemeleri halinde, harita üzerinde simge yerleştirmekten daha ileride olarak, açıklayıcı coğrafi bilgileri de yaratabileceklerdir. Bu, ilgili ontolojiye dayalı olarak yeni yöntemlerle bilginin otomatik kullanılmasını sağlayan Anlamsal Web ile gerçekleştirilmektedir. Örneğin, web sayfalarındaki coğrafik referansları birbirine bağlayarak ve insanların düşünmekten ziyade sezgi yolu ile öğrendiği referansları içeren bilgileri çekerek, bir seyahat rehberi oluşturmak mümkün olabilecektir.

Mekansal ilişkiler, ontolojiler içinde ifade edilmesine rağmen, karmaşık mekansal çıkarımlar yüklemek oldukça zordur. Kalitatif (nitel) mekansal ilişkiler hakkında çıkarsamada bulunulamayacağından dolayı, OWL ile mekansal ilişkilerin ifade edilmesi kısıtlıdır. Daha da açıklamak gerekirse, kalitatif (nitel) mekansal ilişkiler, koordinat tabanlı servislere dayalıdır.

Proje kapsamında gerçekleştirilen veri alışverişi, temel olarak sistemlerin gerçek kapasitelerini yansıtmamaktadır. Ancak, ilk entegrasyon adımı olarak kabul edilebilir ve devamında daha detaylı bilgi alışverişi sağlanabilecektir. Ayrıca, bu tez çalışmasının bir ileri adımı olarak, heterojen sistemler arası değiştirilen verilerin veritabanlarına gömülmesi çalışması yapılabilir. Böylelikle, sadece gösterim amaçlı bir bilgi değişiminden ziyade, değişen bilgiyi alan sistem, bu bilgiyi kendi veritabanına kaydedecektir. Bunun sonucunda, aynı veri omurgasına bağlı sistemlerde veri hattının bir sebepten kaybolması durumunda değiştirilen bilgi istenildiğinde sorgulanabilecektir.

İleri zamanlarda, proje kapsamında elde edilmeye çalışan anlamsal coğrafi verileri, anlamsal olarak sorgulayabilmek amacıyla özelleştirilmiş çıkarsama motoru tasarımı yaratmak da mümkün olabilecektir.

Bu çalışmada kısaca yukarıdaki konulara özenle yer verilmiş olup ontolojilerin gösterimi amaçlı detaylı bir örnek kullanılmıştır. Coğrafi alanın kavramsallaştırmanın en zengin bir şekilde hangi ontoloji dili ile gerçekleştirilebileceği hakkında kritik bir analiz yapılarak özel ilgi alanlarına göre mekansal çıkarsama yapabilecek ve mekansal ilişkileri ortaya koyabilecek şekilde bir sunum yapılmıştır. Bu kapsamda, DAML+OIL ve diğer ontoloji dillerinin yeteneklerinin tam olarak anlaşılabilmesi için; bir gereksinim tanımlaması yapılması, çeşitli prototip ontolojiler geliştirilmesi ve test edilmesi, ayrıca her dilin gereksinimi karşılamadaki yeteneklerinin ölçülmesi gerekli olduğu sonucuna ulaşılmıştır.

Belli başlı sorunlardan birisi, DAML+OIL'in kendi veri tiplerini (Data Types) desteklememesidir. Bunun yerine, XML veri tiplerini kullanmakta, bu da iki veri değeri arasında mantıksal karşılaştırma yapma imkanı vermemektedir. Bunun sonucunda, koordinat verileri üzerinde iki MBR (Minimum Bounding Rectangle) arasındaki "içerir

(contains)” ilişkisinin varlığını test etmek gibi mekansal çıkarsama yapılması zorlaşmaktadır.

Diğer bir sorun ise, araç yazılımı desteği olmamasıdır. Araç yazılımlar, alan uzmanlarının bilgilerini etkili olarak kodlaması için gereklidirler. Günümüzde, Oiled ve Protégé 2000 gibi araçlar yardımıyla DAML+OIL ve OWL ontolojileri yaratılabilmektedir. Aynı zamanda, FACT (Horrocks, 1998), ve RACER (Haarslev et al., 1999) çıkarsama motorlarına bağlantı yapılarak ontolojilerin tutarlılık kontrolu yapılabilmektedir. Bununla birlikte, DAML+OIL ile birlikte kullanılmak üzere ileri bir sorgulama aracının geliştirilmesine de devam edilmektedir. Halihazırda, çoğu sorgulama dilleri (RQL-Karvounarakis et al., 2002) RDF(S) üçlüsü üzerinde çalışmaktadır. Bundan dolayı, DAML+OIL dilinin bütün özellikleri sağlanmamış ve özellikle bilgi tabanlarının ileri seviyede sorgulanmasını gerçekleştirecek çıkarsama motorları henüz gerçekleştirilememiştir.

Web servisleri, RDF editörlerinin ilgili “landmark”ları bulmasına, tanımlama hatalarını kontrol etmeye, uyumu (consistency) kontrol etmek için çıkarımsama motorlarını uygulamaya imkan verirler. Aynı zamanda, ontolojilere dayalı RDF tanımlamalarında belirtilen coğrafik ilişkileri besleyen coğrafik örneklerin çekilmesinde arama motorlarına yardım ederler. Bu yaklaşımın dezavantajlarına bakarsak, başında ölçeklenebilirlik konuları gelir. Çünkü, OWL formatındaki veri kümeleri oldukça büyüktür. Pratik kullanımda büyük veri kümelerinde çalışabilen depolama sistemlerine ve çıkarsama motorlarına ihtiyaç vardır.

Depolama sistemleri için, farklı ontolojilerde tanımlanan özellikler ve sınıflar arasında bağlantı (mapping) ve düzenleme (alignment) hazırlanmalıdır. Bu özellikler kullanılarak, farklı ontolojilere dayalı heterojen örnekler üzerinde çıkarsama (inference) ve muhakeme (reasoning) yapılabilir.

KAYNAKLAR

- Agent Cities**, 2004, DAML+OIL Urban Ontology, <http://iscte.pt/~lhrm/daml/location>.
- Bechhofer S., Stevens R., Horrocks I., Goble C.**, 2001, {OilEd}: {A} Reason-able Ontology Editor for the Semantic {Web}, [citeseer.nj.nec.com /bechhofer01oiled.html](http://citeseer.nj.nec.com/bechhofer01oiled.html).
- Berners-Lee, T.**, 1999, Weaving the Web., New York: Harper.
- Bishr Y.**, 1999, Probing the Concept of Information Communities - A First Step Toward Semantic Interoperability, in Interoperating Geographic Information Systems Kluwer: pp. 55-71.
- Broekstra J., Kampman A., Harmelen F.**, 2002, Sesame: A Generic Architecture for Storing and Querying RDF and RDF Schema, ISWC 2002, LNCS 2342, pp. 54-68.
- Clarke K.**, 2001, Getting Started with Geographical Information Systems, Prentice Hall, 3rd Ed.
- Clemintini E., Di Felice P.**, 1995, A Comparison of Methods for Representing Topological Relationships. Information Sciences 3: 149-178.
- Cowen D., Ehler G., Mackey H.**, 1988, "Design and Implementation of A Spatial Decision Support System for Site Selection, Güney Carolina Üniversitesi.
- Denker Grit, Hobbs J., Martin D., Narayanan S., Waldinger R.**, 2001, Accessing Information and Services on the DAML-Enabled Web, SRI International, Semantic Web Workshop Hongkong, China.

KAYNAKLAR (devam)

- Doerr M.**, 2003, Towards a Core Ontology for Information Integration. Journal of Digital Information.
- Egenhofer M., Herring J.**, 1990, A Mathematical Framework for the Definition of Topological Relationships. Fourth International Symposium on Spatial Data Handling, Zurich, Switzerland, pp. 803-813.
- Esengin, Ö.**, 2004, Coğrafi Bilgi Sistemlerinin Çevresel Etki Değerlendirme İle Entegrasyonu, Uzmanlık Tezi, Çevre Bakanlığı, Ankara.
- ESRI**, 1997, GIS Concepts Kit, Redlands California.
- Extensible Markup Language, (XML)**, <http://www.w3.org/XML>.
- Fonseca F., Egenhofer M.**, 1999, Ontology-Driven Geographic Information Systems. 7th ACM Symposium on Advances in Geographic Information Systems, Kansas City, MO.
- Fonseca F., Egenhofer M., Agouris P., Câmara G.**, 2002, Using Ontologies for Integrated Geographic Information Systems. Transactions in GIS 6(3): 231-257.
- Grosso W., Eriksson H., Fergerson R., Gennari J., Tu S., Musen M.**, 2000, Knowledge Modeling at the Millennium -- The Design and Evolution of Protégé-2000. Proceedings of the 12 th International Workshop on Knowledge Acquisition, Modeling and Mangement (KAW'99), Banff, Canada.
- Gruber, T.**, 1993, A translation approach to portable ontologies, Knowledge Acquisition,5: 199-220.

KAYNAKLAR (devam)

- Gruber, T.**, 1993, Toward Principles for the Design of Ontologies Used for Knowledge Sharing. *International Journal of Human and Computer Studies* 43(5/6): 907-928.
- Haarslev V., Ralf Moller**, 1999, {RACE} System Description, Description Logics, citeseer.nj.nec.com/article/haarslev99race.html.
- Harvey F.**, 1999, Semantic Interoperability: A Central Issue for Sharing Geographic Information. *Annals of Regional Science*, 33 (2) pp. 213-232.
- Hendler J.**, 2001, Agents and the Semantic Web, *Intelligent Systems and Their Applications*, vol. 16, no. 2, pp. 30-37.
- Hernández D., Clemintini E., Di Felice P.**, 1995, Qualitative Distances. *Proceedings of Conference on Spatial Information Theory (COSIT'95)*, A. Frank and W. Kuhn (eds), Semmering, Austria, September 21-23, *Lecture Notes in Computer Science* 988, Springer.
- Heywood, I., Cornelius, S., Carver, S.**, 2002, *An Introduction to Geographical Information Systems*, Prentice Hall, 2nd Ed.
- Hong J., Egenhofer M., Frank A.**, 1995, On the Robustness of Qualitative Distance- and Direction-Reasoning. In *Proceedings of Auto-Carto 12*, Charlotte, North Carolina, pp. 301-310, (eds).
- Horrocks I.**, 1998, Using an Expressive Description Logic: {FaCT} or Fiction, citeseer.nj.nec.com/horrocks98using.html.
- Horrocks I.**, 2002, {DAML}+{OIL}: A Reasonable Web Ontology Language, citeseer.nj.nec.com/579456.html

KAYNAKLAR (devam)

- Horrocks I., Sean Bechhofer, Carole Goble and Agent Cities**, 2004, DAML+OIL Urban Ontology, <http://iscte.pt/~lhrm/daml/location>.
- Jones C.**, 1999, Geographical Information Systems and Computer Cartography, Longman, 2nd Ed.
- Jones C., Alani H., Tudhope D.**, 2001, Geographical Information Retrieval with Ontologies of Place. In Proceedings of COSIT'01, pp. 322-335,
- Jones C.B., Abdelmoty A.I., and Fu Gaihua**, 2003, Maintaining Ontologies for Geographical Information Retrieval on the Web, Ontologies, Databases, and Applications of Semantics for Large Scale Information Systems (ODBASE'03). Proceedings of the OTM Confederated Int. Conf., Springer Verlag, LNCS 2888, pp. 934-951.
- Karvounarakis G., Alexaki S., Christophides V., Plexousakis D., Scholl M.**, 2002, {RQL}: A Declarative Query Language for {RDF}, In the 11th Intl. World Wide Web Conference (WWW2002).
- Koivunen M.R. and Miller E.**, 2001, Semantic Web Kick-off Seminar, Finland.
- Komesli, M.**, 1999, Bölge Seçimi için Gerçekleştirilmiş Coğrafi Bilgi Sistemler Uygulaması, E.Ü. Uluslararası Bilgisayar Enstitüsü, Y.L. Tezi, İzmir.

KAYNAKLAR (devam)

- Komesli M., Özakar B., Tiryakioğlu M., Turhan Ç.,** 2002, Anlamsal Web’de Sorgulama, Ege Üniversitesi Bilgisayar Mühendisliği, İzmir.
- Kuhn W.,** 2003, Semantic Reference Systems. International Journal of Geographic Information Science, 17(5): pp. 405-409.
- Kiyoki Y.,** 1995, A fundamental Framework for realizing semantic Interoperability in a Multidatabase Environment. Integrated Computer Aided Engineering, 2(1) 3- 20.
- McGuinness D.,** 2002, Ontologies Come of Age, Spinning the Semantic Web: Bringing the World Wide Web to Its Full Potential, MIT Press.
- OGC Recommendation Paper,** 2001, Geography Markup Language (GML) 2.0, citeseer.nj.nec.com/422221.html.
- Taştan, H. ve E. Bank,** 1994, Coğrafi Bilgi Sistemlerinde Konuma Bağlı Analizler, 1 nci Ulusal Coğrafi Bilgi Sistemleri Sempozyumu - Bildiriler Kitabı, Sf : 33-52, 1994, KTÜ, Trabzon.
- Tecim V.,** 1997, “A GIS Based Decision Support System for Tourism Planning and Development”, Doktora Tezi, Lanchester Üniversitesi.
- Tecim, V.,** 2001, Coğrafi Bilgi Sistemleri Temel Kavramlar, Uygulama Alanları, DEÜ, İzmir.

KAYNAKLAR (devam)

Ünalır M., Öztürk Ö., 2003, Semantic Web from Metadata Perspective, Ege University Comp.Eng.Dept, Izmir-TURKEY.

W3C, 2001, Semantic Web Introduction, <http://www.w3.org/2001/sw>.

Yomralıoğlu, T., 2000, Coğrafi Bilgi Sistemleri Temel Kavramlar ve Uygulamalar, KTÜ, Trabzon.

<<http://www.cnr.vt.edu/dendro/forsite/2002presentations/gislayers.htm> >, (23.08.2006).

<<http://map.sdsu.edu/images/GIS-fire-1.jpg>>, (23.08.2006).

<<http://www.iiasa.ac.at/Research/LUC/GIS/giswebpage/images/slope.5.gif>>, (22.08.2006).

<http://www.maptell.com/maps/webmap/images/w_h1.gif>, (22.08.2006) .

<http://www.lapanse.com/venise/vues_du_ciel/satellite.jpg>, (22.08.2006) .

<http://www.sbe.hw.ac.uk/research/watermngment/rick_haynes.htm>, (10.08.2006).

<<http://www.opengis.org/techno/abstract.htm>>, (23.08.2006).

<<http://prdownloads.sourceforge.net/jena/Jena-2.3.zip?download>>, (2006).

<<http://www.mip-site.org>>, (2006).

KAYNAKLAR (devam)

- <<http://www.microsoft.com/downloads/details.aspx?FamilyId=262D25E3-F589-4842-8157-034D1E7CF3A3&displaylang=en>>, (2006).
- <<http://protege.stanford.edu>>, (2006).
- <<http://www.mindswap.org/2004/SWOOP>>, (2006).
- <<http://www.mindswap.org>>, (2006).
- <www.opengis.org/techno/abstract.htm>, (Aralık.2006).
- <<http://www.zeroconf.org>>, (2006).
- <<http://www.multicastdns.org>>, (2006).
- <<http://www.w3.org/XML>>, (2006).
- <<http://www.w3.org/RDF>> , (2006).
- <<http://developer.apple.com/darwin/tools/cvs/howto.html>>, (2006).
- <<http://jmdns.sourceforge.net/>>, (2006).
- <<http://www.w3.org/2001/sw/>> , (30.08.2006).
- <http://james.seng.sg/archives/2004/08/27/digital_content_exchange.html>, (31.08.2006).
- <<http://www.w3.org/Protocols>>, (2006).
- <<http://www.w3.org/2000/xp/Group>>, (2006).

EKLER

Ek 1 Coğrafi Ontoloji

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE rdf:RDF [
  <!ENTITY owl "http://www.w3.org/2002/07/owl#">
  <!ENTITY rdf "http://www.w3.org/1999/02/22-rdf-syntax-
ns#">
  <!ENTITY rdfs "http://www.w3.org/2000/01/rdf-schema#">
  <!ENTITY xsd "http://www.w3.org/2001/XMLSchema#">
]>
<rdf:RDF xml:base="&Location;"
  xmlns:owl="&owl;"
  xmlns:rdf="&rdf;"
  xmlns:rdfs="&rdfs;">

  <!-- Ontology Information -->
  <owl:Ontology rdf:about="">
    <rdfs:label rdf:datatype="&xsd:string">Geo
ontology</rdfs:label>
    <owl:versionInfo rdf:datatype="&xsd:string">0.9 $Id:
Location.owl 621 2006-04-17 15:53:08Z $</owl:versionInfo>
  </owl:Ontology>

  <!-- Classes -->
  <owl:Class rdf:about="#Corridor">
    <rdfs:label
rdf:datatype="&xsd:string">Corridor</rdfs:label>
    <owl:disjointWith rdf:resource="#Ellipse"/>
    <owl:disjointWith rdf:resource="#GeoRef"/>
    <owl:disjointWith rdf:resource="#Leg"/>
    <owl:disjointWith rdf:resource="#Point"/>
    <owl:disjointWith rdf:resource="#Polygon"/>
    <owl:disjointWith rdf:resource="#Polyline"/>
    <owl:disjointWith rdf:resource="#Sector"/>
  </owl:Class>

  <owl:Class rdf:about="#Ellipse">
    <rdfs:label
rdf:datatype="&xsd:string">Ellipse</rdfs:label>
    <owl:disjointWith rdf:resource="#Corridor"/>
    <owl:disjointWith rdf:resource="#GeoRef"/>

```

```

    <owl:disjointWith rdf:resource="#Leg"/>
    <owl:disjointWith rdf:resource="#Point"/>
    <owl:disjointWith rdf:resource="#Polygon"/>
    <owl:disjointWith rdf:resource="#Polyline"/>
    <owl:disjointWith rdf:resource="#Sector"/>
  </owl:Class>

  <owl:Class rdf:about="#GeoRef">
    <rdfs:label
rdf:datatype="&xsd:string">GeoRef</rdfs:label>
    <owl:disjointWith rdf:resource="#Corridor"/>
    <owl:disjointWith rdf:resource="#Ellipse"/>
    <owl:disjointWith rdf:resource="#Leg"/>
    <owl:disjointWith rdf:resource="#Point"/>
    <owl:disjointWith rdf:resource="#Polygon"/>
    <owl:disjointWith rdf:resource="#Polyline"/>
    <owl:disjointWith rdf:resource="#Sector"/>
  </owl:Class>

  <owl:Class rdf:about="#Leg">
    <rdfs:label rdf:datatype="&xsd:string">Leg</rdfs:label>
    <owl:disjointWith rdf:resource="#Corridor"/>
    <owl:disjointWith rdf:resource="#Ellipse"/>
    <owl:disjointWith rdf:resource="#GeoRef"/>
    <owl:disjointWith rdf:resource="#Point"/>
    <owl:disjointWith rdf:resource="#Polygon"/>
    <owl:disjointWith rdf:resource="#Polyline"/>
    <owl:disjointWith rdf:resource="#Sector"/>
  </owl:Class>

  <owl:Class rdf:about="#Point">
    <rdfs:comment rdf:datatype="&xsd:string">A location
represented by a single coordinate.</rdfs:comment>
    <rdfs:label
rdf:datatype="&xsd:string">Point</rdfs:label>
    <owl:disjointWith rdf:resource="#Corridor"/>
    <owl:disjointWith rdf:resource="#Ellipse"/>
    <owl:disjointWith rdf:resource="#GeoRef"/>
    <owl:disjointWith rdf:resource="#Leg"/>
    <owl:disjointWith rdf:resource="#Polygon"/>
    <owl:disjointWith rdf:resource="#Polyline"/>
    <owl:disjointWith rdf:resource="#Sector"/>
  </owl:Class>

  <owl:Class rdf:about="#Polygon">
    <rdfs:label
rdf:datatype="&xsd:string">Polygon</rdfs:label>
    <owl:disjointWith rdf:resource="#Corridor"/>

```

```

    <owl:disjointWith rdf:resource="#Ellipse"/>
    <owl:disjointWith rdf:resource="#GeoRef"/>
    <owl:disjointWith rdf:resource="#Leg"/>
    <owl:disjointWith rdf:resource="#Point"/>
    <owl:disjointWith rdf:resource="#Polyline"/>
    <owl:disjointWith rdf:resource="#Sector"/>
  </owl:Class>

  <owl:Class rdf:about="#Polyline">
    <rdfs:comment rdf:datatype="&xsd:string">A location
    represented by a series of connected
    coordinates.</rdfs:comment>
    <rdfs:label
    rdf:datatype="&xsd:string">Polyline</rdfs:label>
    <owl:disjointWith rdf:resource="#Corridor"/>
    <owl:disjointWith rdf:resource="#Ellipse"/>
    <owl:disjointWith rdf:resource="#GeoRef"/>
    <owl:disjointWith rdf:resource="#Leg"/>
    <owl:disjointWith rdf:resource="#Point"/>
    <owl:disjointWith rdf:resource="#Polygon"/>
    <owl:disjointWith rdf:resource="#Sector"/>
  </owl:Class>

  <owl:Class rdf:about="#Sector">
    <rdfs:label
    rdf:datatype="&xsd:string">Sector</rdfs:label>
    <owl:disjointWith rdf:resource="#Corridor"/>
    <owl:disjointWith rdf:resource="#Ellipse"/>
    <owl:disjointWith rdf:resource="#GeoRef"/>
    <owl:disjointWith rdf:resource="#Leg"/>
    <owl:disjointWith rdf:resource="#Point"/>
    <owl:disjointWith rdf:resource="#Polygon"/>
    <owl:disjointWith rdf:resource="#Polyline"/>
  </owl:Class>

  <owl:Class rdf:nodeID="b7">
    <owl:unionOf rdf:parseType="Collection">
      <rdf:Description rdf:about="#Point"/>
      <rdf:Description rdf:about="#Polyline"/>
    </owl:unionOf>
  </owl:Class>

<!-- Datatypes -->
  <rdfs:Datatype rdf:about="&xsd:string"/>

<!-- Annotation Properties -->
  <owl:AnnotationProperty rdf:about="&rdfs:comment"/>
  <owl:AnnotationProperty rdf:about="&rdfs:label"/>

```

```

    <owl:AnnotationProperty rdf:about="#owl:versionInfo"/>

<!-- Datatype Properties -->
    <owl:DatatypeProperty rdf:about="#Altitude">
        <rdfs:comment rdf:datatype="#xsd:string">Altitude above
the surface in meters. Negative numbers imply
depth.</rdfs:comment>
        <rdfs:domain rdf:resource="#GeoRef"/>
        <rdfs:label
rdf:datatype="#xsd:string">Altitude</rdfs:label>
        <rdfs:range rdf:resource="#xsd:string"/>
        <owl:equivalentProperty rdf:resource="#altitude"/>
    </owl:DatatypeProperty>

    <owl:DatatypeProperty rdf:about="#Height">
        <rdfs:comment rdf:datatype="#xsd:string">The height in
meters</rdfs:comment>
        <rdfs:domain rdf:resource="#Leg"/>
        <rdfs:label
rdf:datatype="#xsd:string">Height</rdfs:label>
        <rdfs:range rdf:resource="#xsd:string"/>
    </owl:DatatypeProperty>

    <owl:DatatypeProperty rdf:about="#InnerRadius">
        <rdfs:comment rdf:datatype="#xsd:string">The inner
radius of the sector expressed in
kilometers.</rdfs:comment>
        <rdfs:domain rdf:resource="#Sector"/>
        <rdfs:label
rdf:datatype="#xsd:string">InnerRadius</rdfs:label>
        <rdfs:range rdf:resource="#xsd:string"/>
    </owl:DatatypeProperty>

    <owl:DatatypeProperty rdf:about="#Latitude">
        <rdfs:comment rdf:datatype="#xsd:string">Latitude in
wgs84 decimal degrees</rdfs:comment>
        <rdfs:domain rdf:resource="#GeoRef"/>
        <rdfs:label
rdf:datatype="#xsd:string">Latitude</rdfs:label>
        <rdfs:range rdf:resource="#xsd:string"/>
        <owl:equivalentProperty rdf:resource="#latitude"/>
    </owl:DatatypeProperty>

    <owl:DatatypeProperty rdf:about="#Longitude">
        <rdfs:comment rdf:datatype="#xsd:string">Longitude in
wgs84 decimal degrees</rdfs:comment>
        <rdfs:domain rdf:resource="#GeoRef"/>

```

```

    <rdfs:label
rdf:datatype="&xsd:string">Longitude</rdfs:label>
    <rdfs:range rdf:resource="&xsd:string"/>
    <owl:equivalentProperty rdf:resource="#longitude"/>
</owl:DatatypeProperty>

    <owl:DatatypeProperty rdf:about="#MajorAxis">
    <rdfs:comment rdf:datatype="&xsd:string">The major axis
of the ellipse expressed in kilometers</rdfs:comment>
    <rdfs:domain rdf:resource="#Ellipse"/>
    <rdfs:label
rdf:datatype="&xsd:string">MajorAxis</rdfs:label>
    <rdfs:range rdf:resource="&xsd:string"/>
</owl:DatatypeProperty>

    <owl:DatatypeProperty rdf:about="#MinorAxis">
    <rdfs:comment rdf:datatype="&xsd:string">The minor axis
of the ellipse expressed in kilometers.</rdfs:comment>
    <rdfs:domain rdf:resource="#Ellipse"/>
    <rdfs:label
rdf:datatype="&xsd:string">MinorAxis</rdfs:label>
    <rdfs:range rdf:resource="&xsd:string"/>
</owl:DatatypeProperty>

    <owl:DatatypeProperty rdf:about="#OuterRadius">
    <rdfs:comment rdf:datatype="&xsd:string">The outer
radius of the sector expressed in kilometers. Must be
greater than or equal to InnerRadius.</rdfs:comment>
    <rdfs:domain rdf:resource="#Sector"/>
    <rdfs:label
rdf:datatype="&xsd:string">OuterRadius</rdfs:label>
    <rdfs:range rdf:resource="&xsd:string"/>
</owl:DatatypeProperty>

    <owl:DatatypeProperty rdf:about="#Rotation">
    <rdfs:comment rdf:datatype="&xsd:string">Rotation of
ellipse around center point from north to the major axis.
Positive rotation values indicate clockwise
rotation.</rdfs:comment>
    <rdfs:domain rdf:resource="#Ellipse"/>
    <rdfs:label
rdf:datatype="&xsd:string">Rotation</rdfs:label>
    <rdfs:range rdf:resource="&xsd:string"/>
</owl:DatatypeProperty>

    <owl:DatatypeProperty rdf:about="#StartBearing">
    <rdfs:comment rdf:datatype="&xsd:string">The start
bearing of the sector. 0 = north.</rdfs:comment>

```

```

    <rdfs:domain rdf:resource="#Sector"/>
    <rdfs:label
rdf:datatype="&xsd:string">StartBearing</rdfs:label>
    <rdfs:range rdf:resource="&xsd:string"/>
  </owl:DatatypeProperty>

  <owl:DatatypeProperty rdf:about="#StopBearing">
    <rdfs:comment rdf:datatype="&xsd:string">0 =
north</rdfs:comment>
    <rdfs:domain rdf:resource="#Sector"/>
    <rdfs:label
rdf:datatype="&xsd:string">StopBearing</rdfs:label>
    <rdfs:range rdf:resource="&xsd:string"/>
  </owl:DatatypeProperty>

  <owl:DatatypeProperty rdf:about="#Width">
    <rdfs:comment rdf:datatype="&xsd:string">The width in
kilometers</rdfs:comment>
    <rdfs:domain rdf:resource="#Leg"/>
    <rdfs:label
rdf:datatype="&xsd:string">Width</rdfs:label>
    <rdfs:range rdf:resource="&xsd:string"/>
  </owl:DatatypeProperty>

  <owl:DatatypeProperty rdf:about="#altitude">
    <owl:equivalentProperty rdf:resource="#Altitude"/>
  </owl:DatatypeProperty>

  <owl:DatatypeProperty rdf:about="#latitude">
    <owl:equivalentProperty rdf:resource="#Latitude"/>
  </owl:DatatypeProperty>

  <owl:DatatypeProperty rdf:about="#longitude">
    <owl:equivalentProperty rdf:resource="#Longitude"/>
  </owl:DatatypeProperty>

<!-- Object Properties -->
  <owl:ObjectProperty rdf:about="#Coordinates">
    <rdfs:domain rdf:nodeID="b7"/>
    <rdfs:label
rdf:datatype="&xsd:string">Coordinates</rdfs:label>
    <rdfs:range rdf:resource="#GeoRef"/>
    <owl:equivalentProperty rdf:resource="#coordinates"/>
  </owl:ObjectProperty>

  <owl:ObjectProperty rdf:about="#Legs">
    <rdfs:comment rdf:datatype="&xsd:string">The legs of
the corridor</rdfs:comment>

```

```

        <rdfs:domain rdf:resource="#Corridor"/>
        <rdfs:label
rdf:datatype="&xsd:string">Legs</rdfs:label>
        <rdfs:range rdf:resource="#Leg"/>
    </owl:ObjectProperty>

    <owl:ObjectProperty rdf:about="#LocatedAt">
        <rdfs:comment rdf:datatype="&xsd:string">A physical
location</rdfs:comment>
        <rdfs:label
rdf:datatype="&xsd:string">LocatedAt</rdfs:label>
        <rdfs:range rdf:resource="#Point"/>
        <owl:equivalentProperty rdf:resource="#locatedAt"/>
    </owl:ObjectProperty>

    <owl:ObjectProperty rdf:about="#coordinates">
        <owl:equivalentProperty rdf:resource="#Coordinates"/>
    </owl:ObjectProperty>

    <owl:ObjectProperty rdf:about="#locatedAt">
        <owl:equivalentProperty rdf:resource="#LocatedAt"/>
    </owl:ObjectProperty>
</rdf:RDF>

```

ÖZGEÇMİŞ

- Adı Soyadı** : Murat KOMESLİ
- Doğum Tarihi** : 29.04.1964
- Doğum Yeri** : İskenderun, HATAY
- Medeni Hali** : Evli
- Uyruğu** : T.C.
- Görevi** : Muhabere (Bil.Sis.) Yarbay
Kara Kuvvetleri Karargâhı
Bilgi Sistemler Dairesi
Yazılım Geliştirme Şubesi Kısım Amiri
- Lise** : Kuleli Askeri Lisesi, 1982.
- Lisans** : Kara Harp Okulu,
Harita Mühendisliği Bölümü, 1986.
Orta Doğu Teknik Üniversitesi,
Bilgisayar Mühendisliği Bölümü, Bilgi Sistemleri
Eğitimi, 1996.
- Yüksek Lisans** : Ege Üniversitesi
Fen Bilimleri Enstitüsü
Uluslararası Bilgisayar Anabilim Dalı, 1999.
- Yabancı Dil** : İngilizce

