

**T.C.
YILDIZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

**İDARİ BİRİMLER İÇİN BİR YER ADLARI DİZİN ONTOLOJİSİ GELİŞTİRİLMESİ
VE İSTANBUL ÖRNEĞİNDE BAĞLI VERİ YAKLAŞIMIYLA SUNUMU**

ABDULLAH KARA

**YÜKSEK LİSANS TEZİ
HARİTA MÜHENDİSLİĞİ ANABİLİM DALI
GEOMATİK PROGRAMI**

**DANIŞMAN
DOÇ. DR. VOLKAN ÇAĞDAŞ**

İSTANBUL, 2014

T.C.
YILDIZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

**İDARİ BİRİMLER İÇİN BİR YER ADLARI DİZİN ONTOLOJİSİ GELİŞTİRİLMESİ
VE İSTANBUL ÖRNEĞİNDE BAĞLI VERİ YAKLAŞIMIYLA SUNUMU**

Abdullah KARA tarafından hazırlanan tez çalışması 01.12.2014 tarihinde aşağıdaki jüri tarafından Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü Harita Mühendisliği Anabilim Dalı'nda **YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

Tez Danışmanı

Doç. Dr. Volkan ÇAĞDAŞ
Yıldız Teknik Üniversitesi

Jüri Üyeleri

Doç. Dr. Volkan ÇAĞDAŞ
Yıldız Teknik Üniversitesi

Doç. Dr. Arif Çağdaş AYDINOĞLU
Gebze Teknik Üniversitesi

Prof. Dr. Hülya DEMİR
Yıldız Teknik Üniversitesi

ÖNSÖZ

Bu tez, Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü Harita Mühendisliği Anabilim Dalı'nda yüksek lisans tezi olarak, "İdari Birimler İçin Bir Yer Adları Dizin Ontolojisi Geliştirilmesi Ve İstanbul Örneğinde Bağlı Veri Yaklaşımıyla Sunumu" adıyla hazırlanmıştır.

Tez çalışmam boyunca bana yardımcı olan, benim fikirlerimi dinlemeye değer bulup zaman ayıran, önerileri ile yol gösteren, danışmanım Doç. Dr. Volkan ÇAĞDAŞ'a minnet ve şükran duygularımı ifade etmekle birlikte, teşekkür ederim.

Hiçbir zaman beni geri çevirmeyen, sorduğum saçma sorulara bile sabırla cevap veren ve Web sunumunda bilgisini benden esirgelemeyen Yrd. Doç. Dr. Burak AKPINAR'a; bana konumsal standartları tanıtan, şu anki çalışmalarımın temelini oluşturan bilgileri veren, sevgili hocam Doç. Dr. Arif Çağdaş AYDINOĞLU'na teşekkür ederim.

Annem Semiha ve babam Yüksel'e kararlarımı hep saygı gösterdikleri, bana hep destek verdikleri için teşekkür ederim. Ayberk'e teşekkür ederim.

Parmaklarımın sayısını biraz geçen en yakın arkadaşlarıma teşekkür ediyorum. Hayat sizle daha güzel.

Aralık, 2014

Abdullah KARA

İÇİNDEKİLER

	Sayfa
KISALTMA LİSTESİ.....	vii
ŞEKİL LİSTESİ.....	x
ÇİZELGE LİSTESİ.....	xii
ÖZET.....	xiii
ABSTRACT.....	xiv
BÖLÜM 1	
GİRİŞ	1
1.1 Literatür Özeti	1
1.2 Tezin Amacı	4
1.3 Hipotez.....	4
BÖLÜM 2	
YER ADLARI DİZİNİ	5
2.1 Giriş	5
2.2 Yer Adları Dizini ve Bileşenleri.....	7
2.2.1 Yer Adı.....	8
2.2.2 Detay Sınıfı	9
2.2.3 Konum.....	12
2.2.4 Yer Adları Dizinlerinin İlave Bileşenleri	14
2.3 Küresel Yer Adları Dizinlerine İlişkin Standart Modeller	15
2.3.1 ADL Yer Adları Dizini Modeli	15
2.3.2 Getty Coğrafi Adlar İlişkisel Kavramlar Dizini	17
2.3.3 ISO 19112 Yer Adları Dizini Standardı.....	18

2.3.4	Avrupa Coğrafi Adlar Dizini	20
2.3.5	INSPIRE Coğrafi Adlar Standardı	22
2.4	Uluslararası Yer Adları Dizin Protokolleri	23
2.4.1	ADL Yer Adları Dizini Protokolü	24
2.4.2	OGC Yer Adları Dizini Protokolü.....	24
BÖLÜM 3		
SEMANTİK WEB ve BAĞLI VERİ.....		28
3.1	Bağlı Veri İlkeleri.....	31
3.2	URI Tanımlama İlkeleri	36
3.2.1	Kaynak Tanımlayıcıları	37
3.2.2	Semantik Web’de URI	39
3.2.3	303 ve Hash URI Yaklaşımı.....	41
3.2.4	Küresel URI Modelleme Çalışmaları	45
3.2.5	URI’lerin Bağlantılı Hale Getirilmesi	50
3.3	RDF Veri Modeli	51
3.3.1	RDF Veri Modeli Formatları	57
3.3.2	RDF Oluşturma ve Doğrulama Araçları	60
3.3.3	Coğrafi Veri ve RDF	61
3.4	Ontolojiler	63
3.4.1	Ontoloji Dilleri.....	64
3.4.2	Ontoloji Geliştirme Metodolojileri	72
3.4.3	Ontoloji Geliştirme Araçları	73
3.4.4	Varolan RDF Sözlükleri.....	74
3.5	SPARQL ve GeoSPARQL	77
3.5.1	SPARQL	77
3.5.2	GeoSPARQL.....	81
3.6	RDF Sözlükleri ve Veri Modelleri Arasında Bağlantılar Kurulması	82
3.7	Bağlı Veri Yayımlama Yöntemleri	84
3.8	Semantik Web Kapsamındaki Bağlı Veri Uygulamaları	87
3.9	Coğrafi Verinin Bağlı Veri Olarak Modellenmesi.....	90
BÖLÜM 4		
TÜRKİYE İDARİ BİRİMLER DİZİNİ İÇİN BİR MODEL ÖNERİSİ VE DİZİNİN İSTANBUL ÖRNEĞİNDE BAĞLI VERİ OLARAK YAYIMLANMASI		94

4.1	Mevzuat Analizi	95
4.2	Türkiye İdari Birimler Ontolojisi	101
4.3	Türkiye İdari Birimler Dizininin İstanbul/Beşiktaş Örneğinde Bağlı Veri Sunumu	110
4.3.1	Veri toplama ve analiz	111
4.3.2	İdari Birimler için URI Tanımlama İlkeleri	113
4.3.3	İstanbul İli Beşiktaş İlçesi RDF Dokümanlarının Oluşturulması	116
4.3.4	RDF Dokümanlarının Bağlı Veri Olarak Yayımlanması	118
4.3.5	Türkiye İdari Birimler Dizininin İstanbul İli Beşiktaş İlçesi Bağlı Veri Uygulaması ve Web Sunumu	121
BÖLÜM 5		
SONUÇ VE ÖNERİLER		125
KAYNAKLAR		129
EK-A		
GEOMETRİ ONTOLOJİSİ		138
EK-B		
KONUMSAL İLİŞKİLER ONTOLOJİSİ		142
EK-C		
İDARİ BİRİMLER ONTOLOJİSİ		148
EK-D		
İDARİ BİRİM RDF ÖRNEKLERİ		175
EK-E		
VOID SÖZLÜĞÜ		177
EK-F		
SPARQL SORGU ÖRNEKLERİ		178
ÖZGEÇMİŞ		179

KISALTMA LİSTESİ

ADL	Alexandria Dijital Yer Adları Dizini (Alexandria Digital Library Gazetteer)
API	Uygulama Programlama Arayüzü (Application Programming Interface)
ARE3NA	Yeniden Kullanılabilir INSPIRE Referans Platformu (A Reusable INSPIRE Reference Platform)
ARQ	Jena için bir SPARQL İşlemcisi (A SPARQL Processor for Jena)
ASSCI	Bilgi Değişimi İçin Amerikan Standart Kodlama Sistemi (American Standard Code for Information Interchange)
BBC	Britanya Yayın Kuruluşu (British Broadcasting Corporation)
CBS	Coğrafi Bilgi Sistemi
CORBA	Genel Nesne İstek Aracı Mimari (Common Object Request Broker Architecture)
CSV	Virgülle Ayrılmış Dosyalar (Comma-Separated Values)
CWA	Kapalı Dünya Varsayımı (Close World Assumption)
DAWG	RDF Veri Erişimi Çalışma Grubu (RDF Data Access Working Group)
DCMI	Dublin Temel Meta Veri Girişimi (Dublin Core Metadata Initiative)
DE-9IM	Boyut Olarak Genişletilmiş Dokuz Kesişim Modeli (Dimensionally Extended nine-Intersection Model)
DL	Birinci Seviye Mantık (First Order Logic)
DOLCE	Dilbilimsel ve Kavramsal Mühendislik için Tanımlayıcı Ontoloji (Descriptive Ontology for Linguistic and Cognitive Engineering)
EGN	Avrupa Coğrafi Adlar Dizini (EuroGeoNames)
ETL	Çıkart, Dönüştür, Yükle (Extract, transform, load)
FOAF	Kişisel İlişkiler Sözlüğü (Friend of a Friend)
FTT	İlişkisel Kavramlar Dizinini (Feature Type Thesaurus)
GCS	Yer Adları Dizini İçerik Standardı (Gazetteer Content Standard)
GDB	Coğrafi Veri Tabanı (Geodatabase)
GML	Coğrafi İşaretleme Dili (Geography Markup Language)
GPS	Küresel Konumlama Sistemi (Global Positioning System)
HGK	Harita Genel Komutanlığı
HTML	Zengin Metin İşaret Dili (Hyper-Text Markup Language)
HTTP	Zengin Metin Transfer Protokolü (Hyper-Text Transfer Protocol)
INSPIRE	Avrupa Birliği Konumsal Veri Altyapısı (Infrastructure of Spatial Information in the European Community)

IRI	Uluslararası Kaynak Tanımlayıcı (Internationalized Resource Identifier)
ISA	Avrupa Kamu Yönetimi İçin Birlikte Çalışabilirlik Çözümleri (Interoperability Solutions for European Public Administration)
ISBN	Uluslararası Standart Kitap Numarası (International Standard Book Number)
ISO	Uluslararası Standartlar Organizasyonu (International Organization for Standardization)
ISSN	Uluslararası Standart Seri Numarası (International Standard Serial Number)
JSON	Javascript Nesne Değişkeni (Javascript Object Notation)
KOS	Bilgi Organizasyon Sistemi (Knowledge Organization System)
KVA	Konumsal Veri Altyapısı
LOD	Açık Bağlı Veri (Linked Open Data)
NMCA	Ulusal Harita ve Kadastro Kurumları (National Mapping and Cadastral Agencies)
NUTS	İstatistiki Bölge Birimleri Sınıflandırması (Nomenclature of Territorial Units for Statistics)
OGC	Açık Konumsal Bilgi Konsorsiyumu (Open Geospatial Consortium)
OO	Nesne Yönelimli (Object Oriented)
OS	Birleşik Krallık Harita Kurumu (Ordnance Survey)
OSM	Açık Sokak Haritaları (OpenStreetMap)
OWA	Açık Dünya Varsayımı (Open World Assumption)
OWL	Web Ontoloji Dili (Web Ontology Language)
QGIS	Kuantum Coğrafi Bilgi Sistemi (Quantum Geographic Information System)
RCC8	Alan Bağlantı Matematiği (Region Connection Calculus)
RDF	Kaynak Tanımlama Çatkısı (Resource Description Framework)
RDFa	Özniteliklerle Kaynak Tanımlama Çatkısı (Resource Description Framework in Attributes)
RDFS	Kaynak Tanımlama Çatkı Şeması (Resource Description Framework Schema)
REST	Temsili Durum Transferi (Representational State Transfer)
SEMIC	Semantik Birlikte Çalışabilirlik Komitesi (Semantic Interoperability Community)
SILK	Semantik Web Bağlantı Keşfetme Çatkısı (A Link Discovery Framework for the Web of Data)
SKOS	Basit Bilgi Organizasyon Sistemi (Simple Knowledge Organization System)
SLD	Kartografik Düzenlemeler Servisi (Styled Layer Descriptor)
SOA	Servis Yönelimli Mimari (Service Oriented Architecture)
SOAP	Basit Nesne Erişim Protokolü (Simple Object Access Protocol)
SPARQL	SPARQL Protokol ve RDF Sorgu Dili (SPARQL Protocol and RDF Query Language)
SQL	Yapılandırılmış Sorgu Dili (Structured Query Language)
SUMO	Üst Düzey Birleştirilmiş Ontoloji (Suggested Upper Merged Ontology)
SWRL	Semantik Web Kural Dili (Semantic Web Rule Language)
SWSE	Semantik Web Arama Motoru (Semantic Web Search Engine)
TGN	Getty Coğrafi Adlar İlişkisel Kavramlar Dizini (Getty Thesaurus of Geographic Names)
TUCBS	Türkiye Ulusal Coğrafi Bilgi Sistemi
TÜİK	Türkiye İstatistik Kurumu
UAVT	Ulusal Adres Veri Tabanı

UMBEL	Üst Düzey Eşleşme ve Değişim Katmanı (Upper Mapping and Binding Exchange Layer)
UML	Bütünleşik Modelleme Dili (Unified Modeling Language)
URI	Birörnek Kaynak Tanımlayıcı (Uniform Resource Identifier)
URL	Birörnek Kaynak Konumlayıcı (Uniform Resource Locator)
URN	Birörnek Kaynak Adı (Uniform Resource Name)
USGS	Birleşik Devletler Harita Kurumu (The National Map of the United States)
VOID	Bağlantılı Veri Setleri Sözlüğü (Vocabulary of Interlinked Datasets)
W3C	Dünya Web Konsosiyumu (World Wide Web Consortium)
WFS-G	Web Yer Adları Dizini Harita Servisi (Web Map Service – Gazetteer)
WGS	Dünya Jeodezik Sistemi (World Geodetic System)
WKT	Bilinen Geometri Yazısı (Well-Known Text)
WMS	Web Harita Servisi (Web Map Service)
WMTS	Web Harita Saklama Servisi (Web Map Tile Service)
XML	Genişletilebilir İşaretleme Dili (Extensible Markup Language)
XSD	Genişletilebilir İşaretleme Dili Şema Tanımlaması (Extensible Markup Language Schema Definition)

ŞEKİL LİSTESİ

	Sayfa
Şekil 2.1 1823 tarihli bir yer adları dizininin kapak sayfası.....	6
Şekil 2.2 Sayısal yer isimleri dizininin temel bileşenleri	8
Şekil 2.3 ISO 19107 Geometri paketi sınıf ve özellikleri	13
Şekil 2.4 ADL konumsal veri sınıfları yapısı.....	14
Şekil 2.5 ADL Yer isimleri dizini özet modeli.....	16
Şekil 2.6 Getty TGN modeli	18
Şekil 2.7 ISO 19112:2005 Coğrafi tanımlayıcılarla konumsal referanslama yapısı.....	19
Şekil 2.8 ISO 19112:2005 UML modeli	19
Şekil 2.9 Yeniden şekillendirilmiş ISO 19112	20
Şekil 2.10 EGN kavramsal veri şeması	21
Şekil 2.11 INSPIRE coğrafi isimler veri teması	23
Şekil 2.12 Konumsal Portal Referans Mimarisi Servis Dağılımı	26
Şekil 3.1 Geleneksel Web ve Semantik Web teknolojileri	31
Şekil 3.2 Semantik Web yapısı.....	32
Şekil 3.3 Semantik Web'de konumsal bilgi içeren Bağlı Veri örnekleri.....	35
Şekil 3.4 URI, URL ve URN ilişkisi	37
Şekil 3.5 URL, URI ve IRI ilişkisi	38
Şekil 3.6 Semantik Web'de bir kaynağın URI'si ve tanımlayıcı dokümanı.....	41
Şekil 3.7 303 URI çözümü	43
Şekil 3.8 İçerik uzlaştırma ve hash URI çözümü.....	44
Şekil 3.9 INSPIRE kadastral parsel URI tanımlaması	46
Şekil 3.10 İngiltere kamu sektörü URI modelleri.....	47
Şekil 3.11 INSPIRE ulaşım veri teması örnek URI tanımlamaları	48
Şekil 3.12 HTML ve RDF dokümanları arasında bağlantı.....	51
Şekil 3.13 Bir RDF çizgesi örneği.....	53
Şekil 3.14 RDF'ye uygun XSD tipleri.....	54
Şekil 3.15 RDF 1.1 yeni model formatları	57
Şekil 3.16 Virtuoso Sponger ile RDF'leştirilebilen veri tipleri	60
Şekil 3.17 GML'den RDF'ye dönüşüm.....	62
Şekil 3.18 SKOS formatında bir ilişkisel kavramlar dizini – HTML sunumu	65
Şekil 3.19 OWL 2 semantik hiyerarşisi.....	69
Şekil 3.20 SPARQL ile Örnek RDF sorgulaması	78
Şekil 3.21 Bağlı Veri yayımlama yöntemleri	85
Şekil 3.22 OSM Bağlı Veri uygulaması	89

Şekil 3.23 Topolojik tanımlamalar arasındaki eşitlikler	92
Şekil 4.1 TUCBS coğrafi isim sınıfları.....	104
Şekil 4.2 Türkiye İdari Birimler Ontolojisi sınıf ve ilişkileri.....	107
Şekil 4.3 Protégé editöründeki 'Büyükşehir İl' sınıfı tanımlamaları	108
Şekil 4.4 İdari birim, geometri ve konumsal ilişkiler ontolojileri.....	109
Şekil 4.5 Ontolojilerin yapısal doğruluk testi.....	110
Şekil 4.6 Tez kapsamında belirlenen pilot bölge	112
Şekil 4.7 Fuseki'ye veri yükleme	118
Şekil 4.8 Fuseki ile idari birim RDF'lerinin sorgulanması.....	119
Şekil 4.9 İdari birim dizini Fuseki sorgulama sonucu.....	120
Şekil 4.10 Türkiye idari birimler dizini Web sitesi ontoloji bölümü.....	121
Şekil 4.11 Türkiye idari birimler dizini Web sitesi liste bölümü	122
Şekil 4.12 İdari birim Web sunumlarının Bağlı Veri olduğunu doğrulaması.....	123
Şekil 4.13 Abbasağa mahallesi için hazırlanan Web sunumu	124

ÇİZELGE LİSTESİ

	Sayfa
Çizelge 2.1 İlişkisel kavramlar dizini ve ontoloji arasındaki farklar.....	10
Çizelge 2.2 Web'deki yer isimleri dizinlerine ilişkin şemalar.....	11
Çizelge 3.1 HTML ve RDF dokümanlarında yer alanı örneği.....	39
Çizelge 3.2 HTTP protokolleri ile 303 URI'sine erişim.....	42
Çizelge 3.3 RDF/XML örneği.....	55
Çizelge 3.4 RDFa örneği.....	58
Çizelge 3.5 RDF Turtle örneği.....	59
Çizelge 3.6 RDF N-Triples örneği.....	59
Çizelge 3.7 SKOS formatında bir ilişkisel kavramlar dizini – SKOS sunumu.....	64
Çizelge 3.8 Bir RDFS örneği.....	67
Çizelge 3.9 Örnek RDF turtle verisi.....	79
Çizelge 4.1 Türkiye'deki mülki ve mahalli idare birimleri.....	101
Çizelge 4.2 'Büyükşehir İl' sınıfı ve bazı ilişkileri.....	108
Çizelge 4.3 Çalışma kapsamında kullanılan URI yapıları.....	115
Çizelge 4.4 İdari Birim RDF/XML örneği.....	117

İDARİ BİRİMLER İÇİN BİR YER ADLARI DİZİN ONTOLOJİSİ GELİŞTİRİLMESİ VE İSTANBUL ÖRNEĞİNDE BAĞLI VERİ YAKLAŞIMIYLA SUNUMU

Abdullah KARA

Geomatik Mühendisliği Kamu Ölçmeleri Anabilim Dalı

Yüksek Lisans Tezi

Tez Danışmanı: Doç. Dr. Volkan ÇAĞDAŞ

Sayısal coğrafi yer adları dizinleri, Web temelli haritacılık servisleri ve konumsal arama motorları gibi uygulamaların temel yapı taşı olarak görülmektedir. Yerel ve küresel coğrafi yer adları dizinleri oluşturmaya yönelik son girişimler, dizinlerin Semantik Web (Semantic Web – Anlamsal Web) dili olan Kaynak Tanımlama Çatısı (Resource Description Framework – RDF) ile temsili ve Bağlı Veri (Linked Data) yaklaşımı ile sunumuna odaklanmıştır. Bu çalışmada, idari birimlere özgü bir coğrafi yer adları dizininin Semantik Web ve Bağlı Veri yaklaşımıyla sunumuna olanak sağlayacak Türkiye İdari Birimler Ontolojisi geliştirilmiş ve İstanbul İli, Beşiktaş İlçesi örneğinde ontoloji tanımlamalarına uygun üretilen RDF dokümanları Bağlı Veri olarak sunulmuştur. Tasarlanan ontoloji, idari birimlerin adları, konumları, öznitelikleri ve birimler arasındaki konumsal ilişkileri (mereolojik ve topolojik ilişkiler), Web Ontoloji Dili (Web Ontology Language – OWL) ile ifade etmektedir. İdari birimler dizini, idari birimlerle ilgili tüm RDF veri setlerinin (örn., nüfus, vergi, kamu hizmetleri, seçim, istatistik) Web yoluyla entegrasyonunu sağlayacaktır. Bu yaklaşım, ilave sorgulama yapmadan bir idari birimle ilgili tüm verilere Web yoluyla erişimi etkinleştirecektir.

Anahtar Kelimeler: İdari Birimler, Yer Adları Dizini, Semantik Web, Bağlı Veri, Ontoloji, RDF, OWL.

ABSTRACT

DEVELOPMENT A GAZETTEER ONTOLOGY FOR ADMINISTRATIVE UNITS AND ITS LINKED DATA REPRESENTATION IN ISTANBUL CASE

Abdullah KARA

Department of Geomatics Engineering

MSc. Thesis

Adviser: Assoc. Prof. Dr. Volkan ÇAĞDAŞ

Digital gazetteers are seen as the basic building blocks of the Web-based mapping services and spatial search engines. The most recent local and global gazetteer development initiatives have focused on representing gazetteers as Linked Data, by using the Resource Description Framework (RDF). In the present study, an ontology model, namely Turkish Administrative Units Ontology is developed, RDF documents are produced and published as Linked Data in the example of İstanbul, Beşiktaş District. The developed ontology, represented by Web Ontology Language (OWL), specifies Turkish administrative units, their locations and properties, as well as spatial relationships (i.e. mereological and topological) between these units. The Linked Data representation of administrative units enables integration of all RDF datasets (e.g. population, taxation, public services, statistics) prepared by different public organizations for the same administrative units. This approach facilitates accessing all administrative unit related information via Web, without making any additional search.

Keywords: Administrative Units, Gazetteer, Semantic Web, Linked Data, Ontology, RDF, OWL.

YILDIZ TECHNICAL UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

1.1 Literatür Özeti

Yer adları dizinleri (gazetteer); coğrafi yerlerin adları, sınıfları, konumları ve coğrafi yerler ile ilgili ek bilgileri içeren konumsal sözlüklerdir. Sayısal formatta yayımlanan yer adları dizinleri özelleşmiş bir Coğrafi Bilgi Sistemi (CBS) olarak değerlendirilmekte [1]; Web temelli haritacılık servisleri ile konumsal arama motorları gibi uygulamaların yapı taşı olarak görülmektedir [2]. Bir yer adları dizininin; yer adı, detay sınıfı ve coğrafi konum olmak üzere literatürde kabul görmüş üç temel bileşeni vardır [3], [4], [5], [6], [7], [8], [9], [10]. Ancak, yer adları dizinleri ilave bileşenleri veya öznitelikleri de içerebilir (örn., alternatif adlar, konumsal ilişkiler, zamansal faktörler ve referans kaynakları) [4], [6].

Günümüze dek çok sayıda yerel ve küresel yer adları dizini yayımlanmıştır. Genellikle basılı formatta yayımlanan dizinler, 1990'larda yerini bilgisayar sistemleri tarafından erişilebilen sayısal dizinlere bırakmıştır [9]. Son yıllarda uluslararası ölçekte çevrimiçi erişimin olanaklı olduğu çeşitli sayısal dizin oluşturma projeleri yürütülmektedir. Bunların başında, Getty Coğrafi Adlar İlişkisel Kavramlar Dizini¹ (Getty Thesaurus of Geographic Names – TGN), Alexandria Dijital Yer Adları Dizini² (Alexandria Digital Library Gazetteer – ADL), EuroGeoNames³ ve Geonames⁴ gelmektedir. Bu sayısal

¹ <http://www.getty.edu/research/tools/vocabularies/tgn/>

² <http://alexandria.ucsb.edu/>

³ <http://www.eurogeographics.org/eurogeonames>

dizinler Geniřletilebilir İřaretleme Dili (Extensible Markup Language – XML) temelli iřaretleme řemaları ile modellenmiřtir. Ayrıca, yer adları dizinlerinin standartlařtırılmasına yönelik řalıřmalar da yřrřtřlmektedir (řrn., ISO 19112 Yer Adları Dizini Standardı, INSPIRE Coęrafi Adlar Standardı) [3], [7]. Yer adları dizinleri oluřturulmasına yönelik son giriřimler (řrn., Geonames), dizinlerin Semantik Web dilli olan Kaynak Tanımlama řatķısı (Resource Description Framework - RDF) ile modellenmesine ve Baęlı Veri yaklařımı ile sunumuna odaklanmıřtır [10], [11], [12].

Semantik Web, Berners-Lee (1998) tarafından, Web’de sunulan verilerin bilgisayarların da anlayabileceęi bir yapıda biçimlendirilmesi amacıyla geliřtirilmiř bir kavramdır [13]. Semantik Web yapısının önemli bir yapı tařı olan Baęlı Veri yaklařımı ise veri setlerinin Web mimarisi řzerinde URI’lerle tanımlanmasını, RDF veri modeliyle temsil edilmesini ve RDF linkleriyle baęlanmasına iliřkin en iyi uygulamaları tanımlar [14]. Bu yaklařım, farklı kaynaklardaki veri setlerinin etkin biçimde iliřkilendirilmesini saęlar [15]. Baęlı Veri yaklařımında URI’ler geręek dřnya nesnelerini ve bunların Web kaynaklarını ifade etmek iin kullanılır [16]. URI’ler ile adlandırılan řeyler hakkındaki veriler RDF ile modellenir. RDF veri modelinin amacı, Web kaynaklarının tanımlanması iin kavramsal bir çereve sunarak, heterojen veri yapılarını dřzenlemek ve Web kaynaklarının herhangi bir alandan (domain) baęımsız tanımlanmasını saęlamaktır [14], [17]. RDF dokřmanlarının oluřturulmasında RDF sřzlřkleri ve ontolojilerden yararlanılır.

Herhangi bir alana řzgř terimler ve bu terimler arasındaki iliřkiler RDF sřzlřkleri ve ontolojilerle sunulur [14]. Ontoloji, kavramsallařtırmanın formel ve aık bir tanımlamasını sunar [18]. Kavramlar ve kavramlar arasındaki iliřkiler farklı tanımlama dřzeylerine sahip ontoloji dilleri ile ifade edilebilir. Bunlar; Basit Bilgi Organizasyon Sistemi (Simple Knowledge Organization System – SKOS), Kaynak Tanımlama řatķı řeması (Resource Description Framework Schema – RDFS) ve Web Ontoloji Dili’dir (Web Ontology Language – OWL) [19], [20], [21], [22], [23]. Bu standart ontoloji dilleri iinde en geniř tanımlamaları OWL sunar.

⁴ <http://www.geonames.org/>

Web’de RDF formatında sunulan veri setlerini sorgulamak için SPARQL Protokol ve RDF Sorgu Dili (SPARQL Protocol and RDF Query Language – SPARQL) kullanılır [24], [25]. OGC tarafından ek protokollerle genişletilen GeoSPARQL, SPARQL sorgulama diline konumsal sorgulama yetenekleri katmıştır [26].

Ülkemizde yer adları dizini kapsamında, Harita Genel Komutanlığı (HGK) tarafından ‘1/250.000 Ölçekli Türkiye ve Çevresi Coğrafi Ad Dizini’ ile bu dizine idari birim detayı açısından veri sağlayan ‘Türkiye Yerleşim Yerleri Veri Tabanı’ geliştirilmiştir. 1/250.000 Ölçekli Türkiye ve Çevresi Coğrafi Ad Dizini, Türkiye ve çevresine ait yerleşim yerleri ile topografik ve hidrografik detayların; adlarını, türlerini ve coğrafi koordinatlarını basılı formatta sunmaktadır. Türkiye Yerleşim Yerleri Veri Tabanı ise yerleşim yerlerinin; ID numarası, resmi yer adı, 1/25.000 ölçekli paftası ve grid numaraları, coğrafi koordinatları ve yüksekliğine ilişkin verileri içermektedir. Veri tabanı, topografik harita üretimi için yapılan arazi çalışmaları ve İçişleri Bakanlığının idari birimlerdeki değişikliklere ilişkin resmi yazılarından yararlanılarak güncellenmektedir. Kullanıcılar HGK Web sayfası üzerinden veri tabanında yerleşim yeri sorgulayabilir; sorgulama sonucunda yerleşim yeri ile ilgili yukarıda sıralanan verilere erişip, yerleşim yerinin (nokta) konumunu harita üzerinden görebilirler [27]. İçişleri Bakanlığı da idari birimlerin basit bir listesini sunan ‘Mülki İdare Birimleri Envanteri’ni oluşturmuştur. Bu envanter; idari birim adlarını basit bir liste biçiminde sunmakta, Web ara yüzü ile sorgulanan birimin bağlı olduğu birim adını ve türünü göstermektedir. Ayrıca yetkili kullanıcılar, sorguladıkları birimlerin konumlarını bir harita üzerinde görebilmektedirler. Ancak bu ulusal dizin ve envanterler; yer adları dizinlerinin temel bileşenlerine kısmen sahip olmakla birlikte, bazı yetersizlikler göstermektedir: (1) Dizinler; detayların alternatif adlarına, zamansal faktörlere ve referans kaynaklarına ilişkin öznitelikleri sunmamaktadır. (2) Çoklu doğru veya alan ile temsil edilmesi gereken idari birimler, nokta ile temsil edilmektedir. (3) Dizinlerde, idari birimlerin sınırlarına ilişkin konum verisi bulunmamaktadır. (4) Dizinler, detaylar arasındaki topolojik ilişkileri tanımlamamaktadır.

1.2 Tezin Amacı

Bu çalışma, yer adları dizinlerinin önemli bir bileşeni olan mülki (il, ilçe, bucak) ve mahalli idari birimleri (il özel idaresi; büyükşehir, il, ilçe, belde belediyeleri; köy ve mahalleler) detayına ülkemiz ölçeğinde odaklanmaktadır. Tezdeki ilk amaç, ulusal ve küresel yer adları dizin çalışmalarının inceleyip bu alanda çıkarımlar yapmaktır. Kurulması bu çalışma ile önerilen Türkiye İdari Birimler Dizini için mevzuata ve ulusal, küresel standartlara uygun bir İdari Birim Ontolojisi geliştirilmesi tezin ikinci amacıdır. Geliştirilen ontolojiye ve Semantik Web kapsamındaki Bağlı Veri yaklaşımının tüm ilkelerine uygun olarak Türkiye İdari Birimler Sayısal Dizininin İstanbul / Beşiktaş örneğinde pilot uygulamasının gerçekleştirilmesi tezdeki üçüncü amaçtır.

1.3 Hipotez

Çalışma kapsamında hipotez şu şekilde kurulmuştur:

- Bu tez ile geliştirilen Türkiye İdari Birimler Ontolojisi; idari birimlerin adları, öznitelikleri ve bu birimler arasında mevzuatla belirlenmiş açık ve zımni ilişkilerin formel biçimde tanımlanmasına olanak verir.
- Bu tez ile geliştirilmesi önerilen, pilot uygulaması Bağlı Veri yaklaşımıyla sunulan Türkiye İdari Birimler Dizini; adları ve sınırları sıklıkla değiştirilen idari birimleri kalıcı URI'lerle tanımlar, RDF veri modeli ile ifade eder, uygun geometri ile temsil eder, idari birimler arasındaki mereolojik ve topolojik ilişkileri açıklığa kavuşturur.

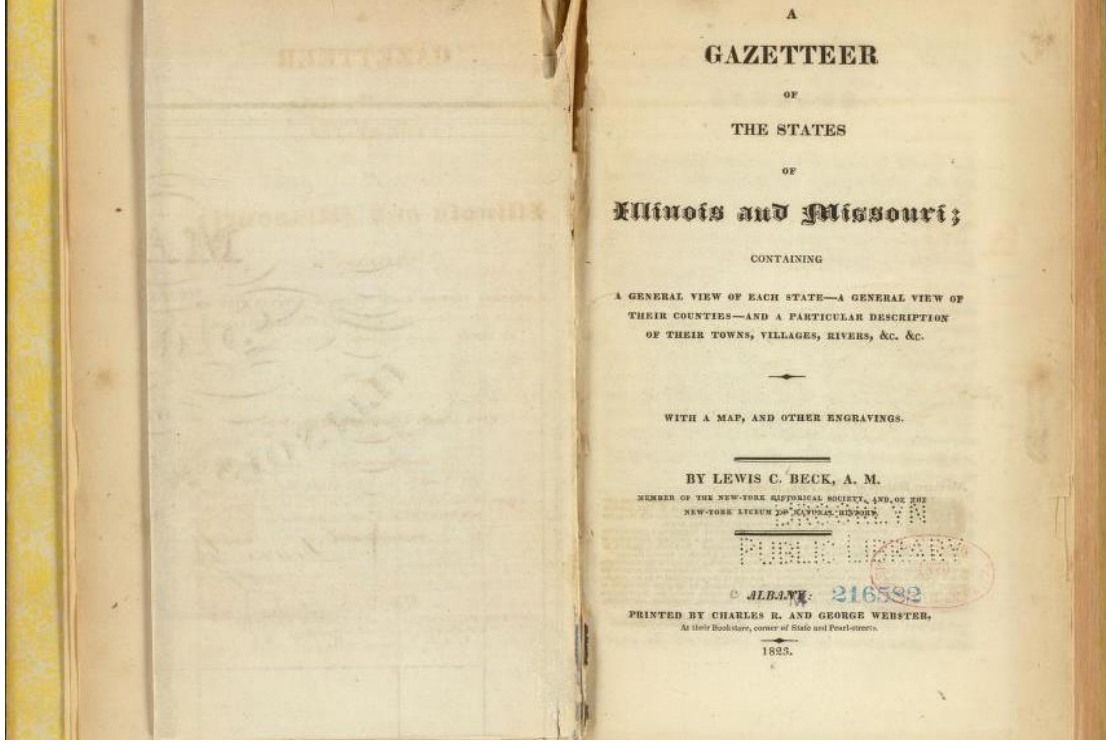
BÖLÜM 2

YER ADLARI DİZİNİ

2.1 Giriş

İlk çağlardan itibaren insanlar genellikle yerin doğal özelliklerinden ve/veya bir olaydan etkilenecek coğrafi detayları adlandırmışlardır. Yer adlarının sayısının zamanla artması ve yer adlarıyla birlikte o yerin bazı niteliklerini de öğrenme isteği; yer adlarını toplama, listeleme, yayınlama gereksinimini doğurmuştur. Yer adları dizini (gazetteer) olarak bilinen derlemeler bu gereksinimi karşılamak üzere geliştirilmiştir. Daha resmi bir anlatımla, bir yer adları dizini; coğrafi yerler ile ilgili formel bilgileri içeren bir araçtır [5] ve “*Adı X olan yer nerededir?*” ya da “*Y konumunda hangi detaylar bulunmaktadır?*” gibi sorgulamalara cevap verir.

Modern yer adları dizinine benzer yayımların tarihi milattan öncesine dayanır. Bilinen ilk dizin, milattan sonra altıncı yüzyılda hazırlanmış olan “Ethnica” isimli coğrafi sözlüktür [28], [29]. Daha yakın döneme bakıldığında, Çinliler on yedinci yüzyıldan, İngilizler de on dokuzuncu yüzyıldan itibaren geleneksel yer adları katalogları hazırlamaya başlamışlardır [30]. “Yer adları dizini” (gazetteer) terimi modern anlamda ilk kez on sekizinci yüzyılda Lawrence Echard tarafından kullanılmıştır [31]. On sekizinci yüzyıldan günümüze değin yer adları dizinlerinin üretimi ve kullanımı artarak sürmüştür. Şekil 2.1, Lewis ve Beck tarafından Illinois ve Missouri için 1823 yılında hazırlanan bir yer isimleri dizininin kapak sayfasını göstermektedir [32].



Şekil 2.1 1823 tarihli bir yer adları dizininin kapak sayfası [32]

Yer adları dizinleri, yer adları ve bunların kabul görmüş imlasını belgeleyen referans kataloglar olarak üretilmeye başlanmıştır [8]. On dokuzuncu ve yirminci yüzyıllarda çok sayıda ulusal, bölgesel ve uluslararası yer adları dizini kartografik niteliği düşük olan monografiler biçiminde yayınlanmıştır [9]. Ancak, günümüzdeki dizinler, yer adlarının hiyerarşik listesi olarak basılmış erken dönem örneklerinden oldukça farklılaşmaktadır [4]. CBS'lerin gelişmesi ve Web teknolojilerinin artan kullanımı; yer adlarına ilişkin Web tabanlı sorgulama gereksinimlerini karşılayacak sayısal dizin modellerinin ortaya çıkmasına neden olmuştur.

Web kullanıcıları, arama motorları ve Web tarayıcıları aracılığıyla birçok bilgiye kolaylıkla ulaşabilmektedir. Bilgilerin önemli bir bölümünün coğrafi alanla ilişkili olmasından ötürü kullanıcılar sorgulamalarında konumu tanımlamak isterler [33]. Günümüzde Web üzerinden sunulan çoğu CBS uygulamasında konum, coğrafi koordinatlarla temsil edilir. Ancak kullanıcılar konumla ilgili bilgilere erişmek için koordinat yerine yer adları dayanaklı konumsal referans sistemini tercih eder [5]. Bu nedenle yer adları, konumla ilgili bilgiye erişimde temel unsur olarak görülür. Geleneksel olarak basılı kitap biçiminde veya topoğrafik haritaların eki olarak

hazırlanmış dizinler; günümüzde bu gereksinimi karşılamak üzere dijital formatta yayımlanmaktadır. Yer adları dizinleri; coğrafi yerlerin adlarını, konumunu ve yer ile ilgili ek bilgileri içeren, Web ve CBS gibi teknolojilerle yayımlanan, sayısal konumsal sözlüklere evrilmiştir. Sayısal yer adları dizinleri günümüzde, özelleşmiş bir CBS olarak değerlendirilmekte [1] ve Web temelli haritacılık servisleri ile konumsal arama motorları gibi uygulamaların yapı taşı olarak görülmektedir [2]. Ancak, sayısal yer adları dizinleri, kullanım amacı ve sunum yönünden klasik CBS'den farklılaşmaktadır. Klasik CBS uygulamalarında yer adlarına bir katman olarak davranılmış ve yer adları ilgili coğrafi detayı temsil amacıyla kullanılmıştır. Başka bir deyişle, CBS'de vurgulanan, yer adlarından çok, o nesnenin coğrafi konumdur [8]. Oysa, yer adları dizinlerinin odağında yer adları bulunur; coğrafi referanslama (geo-referencing) ile addan o yerin konumuna ulaşılır [5].

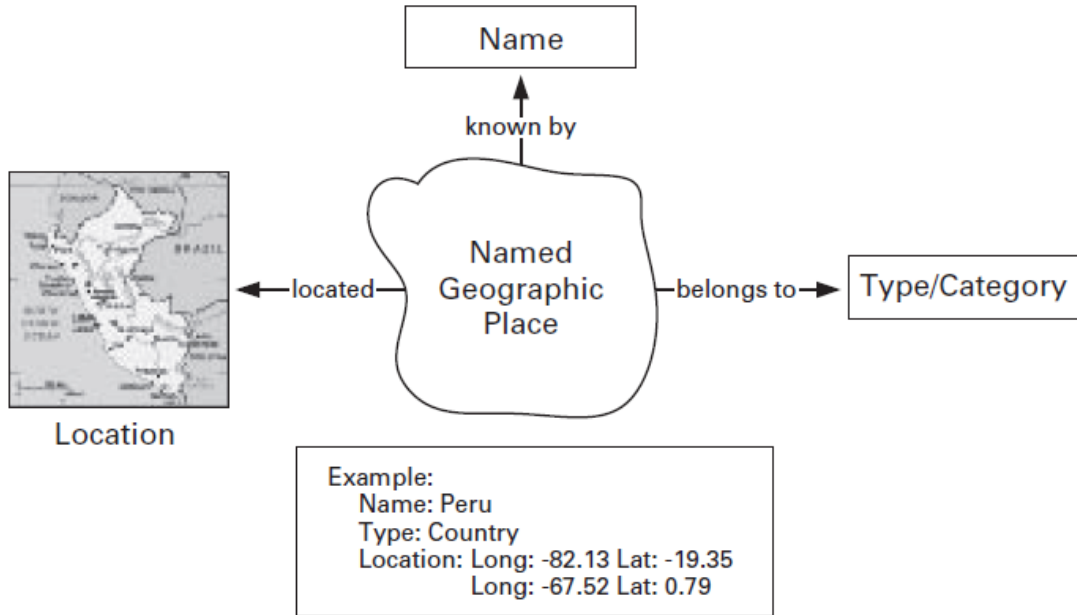
2.2 Yer Adları Dizini ve Bileşenleri

Yer adları dizinleri, insan söylemi (human discourse) ve CBS dünyası arasındaki arayüzü biçimlendirir [8]. Bir yer adları dizininin literatürde kabul görmüş üç temel bileşeni vardır: **(1)** Yer adı (N), **(2)** Detay sınıfı (T) ve **(3)** Coğrafi konum (F) [3], [4], [5], [6], [7], [8], [9], [10].

Goodchild ve Hill (2008), yer (place) ve coğrafi detay (feature) terimlerinin eş anlamlı kullanılmasına karşın, farklı kavramları ifade ettiklerini belirtmişlerdir [8]. Onlara göre coğrafi detaylar, kesin coğrafi konumu ve sınırları belirli olan, doğadaki somut fiziksel nesnelerdir. Yer ise coğrafi detayı kapsamakla birlikte, belirli bir zaman süresince bir topluluk tarafından tanımlanan, belirsiz konumları ve sınırları kapsayan soyut bir terimdir [34], [35]. Özetle, yer terimi tüm coğrafi varlıkları tanımlarken, coğrafi detay terimi sınırları belirli yerleri tanımlamaktadır. Yer adları dizinlerinde hem yer hem de coğrafi detay bulunabilir.

Bir yer adları dizini, içerdiği üç temel bileşenle, iki temel işlevi sağlar. Bu işlevlerden birincisi, yer adı ve coğrafi konum arasındaki ($N \rightarrow F$) eşleşmenin; ikincisi ise yer adı ile detay sınıfı arasındaki ($N \rightarrow T$) eşleşmenin sağlanmasıdır [2], [8], [35]. Şekil 2.2'de bir yer adları dizininin sahip olması gereken temel bileşenler, bir dizin girdisi örneğinde sunulmuştur [4]. Burada, konumu olan bir coğrafi detayın bir adla tanımlandığı ve bu

detayın bir liste, sözlük veya ontoloji ile organize edilen bir detay sınıfına ait olduğu görselleştirilmiştir.



Şekil 2.2 Sayısal yer isimleri dizininin temel bileşenleri [4]

2.2.1 Yer Adı

Yer adı, toponomi (toponym) veya coğrafi isim; yeryüzündeki bir yeri, olguyu ya da alanı tanımlayan veya kasteden, kültürel olarak kabul görmüş, resmi ya da geleneksel bir isimdir [5]. İdari birimler, dağ, nehir gibi coğrafi nesneler ve insan yapımı coğrafi detaylar (man-made features) yer adlarına sahip olabilir. Özetle, bir konum ifade eden somut nesneler, olaylar veya soyut kavramlar da yer adı ile ifade edilebilir [6].

López-Pellicer vd. (2007) yer adlarını farklılaştıran öğeleri; dilbilimsel etkenler (dil, alfabe, telaffuz, kelime ekleri), kartografik etkenler, kaynak referansları, eşsiz tanımlayıcı ve alternatif adlar olarak sıralamıştır [5]. Avrupa Birliği Konumsal Veri Altyapısı (Infrastructure of Spatial Information in the European Community – INSPIRE) Coğrafi İsimler Veri Teması (2007) da, daha geniş kapsamda, ancak benzer bir yaklaşımla yer adlarına ilişkin kavramsal bir model sunmuştur [7].

Yer adları değişik dillerde, değişik formatlarda ifade edilebilir. Bir yer adı her dilde farklı ifade edilebileceği gibi (örn., Türkiye / Turkey); farklı dillerde aynı biçimde kullanılabilir (örn., Ankara / Ankara). Yine farklı ulusların farklı alfabeler kullanması (örn., Grek

alfabesi, Rus Alfabesi); yer adının yerel ve uluslararası düzeyde farklı söylenişlere ve farklı yazılışlara sahip olabilmesine neden olur. Bu nedenle, bir yer adının küresel veya ulusal düzeydeki kullanımında dil faktörü göz önünde tutulur [4], [7].

Bir yer, bir veya birden fazla resmi veya resmi olmayan (geleneksel) isimle adlandırılabilir. Bu durumda, bir adın öncelikli ad olarak seçilmesi ve diğer adların alternatif ad olarak o yeri ifade etmesi sağlanır. Seçilen öncelikli yer adı, genellikle o yerin coğrafi tanımlayıcısı olarak kullanılır. Bunun yanında yer adları dizinlerinde, her yer adını eşsiz bir tanımlayıcıyla ifade etmek de mümkündür.

Dizinler, yer adları arasındaki ilişkileri tanımlayacak biçimde organize edilir. Dizinler, adlandırılmış yerler (named place) hakkındaki bilgileri organize eden özel bir Bilgi Organizasyon Sistemi (Knowledge Organization System - KOS) olarak da düşünülebilir [4]. KOS'larda, yer adları arasında tanımlanmış sistematik ilişkiler yardımıyla bağlantılar kurulabilir. Ancak, bu ilişkiler doğrudan yer adlarına eklenen bazı tanımlayıcılarla da sağlanabilir. Örneğin: İstanbul İli, Beşiktaş İlçesi vd.

Yer adlarının ifade ettiği sınırların belirlenmesi, belirsiz yerlerin sınırlarının modellenmesi, bu sınırları ifade edecek geometriler kapsamında yer adlarının haritaya nasıl yerleştirileceği; günümüzde halen tartışmalı olan konulardan bazılarıdır. İnsanların günlük hayatta konumları tanımlamak için kullandığı yer adları, dizinlerde ifade edilen yerlerle tamamen çakışmamakta veya uyuşmamaktadır. Bu yüzden, resmi sınırı olmayan, birçok belirsiz (vague) ve yerel (vernacular) yer adı vardır (örn., Güney Fransa, Orta Batı, Toros Dağları) [36]. Belirsiz yerleri tanımlamak için farklı yaklaşımlar uygulanır: Sınırların insanlara sorarak belirlenmesi [37]; yakın yer adlarından yararlanarak konumu açıklama [38] bu yaklaşımlardan sadece birkaçıdır.

2.2.2 Detay Sınıfı

Kimi araştırmacılar, yer adı ve konumunu dizinlerin temel bileşeni olarak görmekte; o yerin sınıfını veya detay türünü göz ardı etmektedirler. Ancak, Hill'in (2006) belirttiği gibi, yer adlarının sınıflandırılması, dizinlerin sorgulama ve veriye erişim yeteneklerini geliştiren önemli bir ögedir [4]. Örneğin, bir dizindeki il, ilçe vb. gibi idari detayların bir sınıflandırma sistemi ile organize edilmesi; kullanıcıların bu sınıftaki tüm idari birimlere

erişimini, bu birimler arasındaki ilişkilerin açığa çıkarılmasını kolaylaştırır. Yer adları dizinlerinde *nedir* yer ismi, *nerededir* konum sorularına yanıt olabilirken; nesne toplulukları *detay sınıfları* aracılığıyla bulunabilmektedir [6]. Coğrafi detaylar farklı yaklaşımlarla sınıflandırılabilir.

Çizelge 2.1 İlişkisel kavramlar dizini ve ontoloji arasındaki farklar [2]

Karşılaştırma ölçütü	İlişkisel Kavramlar Dizini	Ontoloji
Amaç	Bilgiye erişim ve bilgiyi yapılandırma	Sonuç çıkarma ve mantıklandırma, bilgiye erişim
Yöntem	Jenerik, parça-bütün veya örnekleme hiyerarşisi	'is-a' ilişkisi
İlişkiler	Sınırlı sayıda ilişki	İsteğe bağlı sayıda ilişki
Düzyey	Kavramlar terimlerle temsil edilir.	Kavramlar tanımlanır.
Semantik	Formel semantik yoktur	Formel semantik

Yer adları dizinindeki yerler, coğrafi detay sınıflarını (feature class) mantıksal şemalarla organize eden KOS'larla şekillendirilebilir [4]. KOS'lar; *liste*, *taksonomi*, *ilişkisel kavramlar dizini* (thesauri) veya *ontoloji* gibi farklı düzeylerde olabilir [2], [4], [5]. Bu seçeneklerden en basit yapıda olanı listedir. Taksonomi basit bir hiyerarşik yapı sunarak, sadece “bir türüdür” ilişkisini tanımlar [38]. İlişkisel kavramlar dizini ise sabit sayıda ilişkinin kullanıldığı kontrollü bir sözlüktür (controlled vocabulary). Taksonomiden farklı olarak, parça-bütün (partonomy) ilişkilerinin, hiyerarşik olmayan ilişkiler (associative) ile kavramların tercih edilen ve edilmeyen terimlerinin (preferred – nonpreferred terms) tanımlanmasını sağlar [2]. Ontolojide ise taksonomi ve ilişkisel kavramlar dizininden farklı olarak, kavramlar arasındaki karmaşık ilişkilerin tanımlanması sağlanır; tanımlanan ilişkiler mantıksal çıkarımlarla (reasoning) doğrulanabilir. Çizelge 2.1’de gösterildiği gibi, ontolojiler ile sınırsız sayıda ilişki kurulabilmekte, bu ilişkilerin mantıksal tutarlılığı denetlenebilmekte, kavramlar ve

kavramlar arasındaki ilişkiler semantik olarak ilişkilendirilebilmektedir. Ontolojiler; taksonomiler ve ilişkisel kavramlar dizinlerinin aksine, yer adları dizinlerini en zengin biçimde yapılandırabilecek bilgi organizasyon şemalarıdır. EDINA [2007], bir ilişkisel kavramlar dizini çoklu coğrafi detay sınıf ilişkileri kurmanın hiyerarşik yapıyı bozduğu ve arama operasyonlarını zorlaştırdığı ifade edilmiştir [6].

Günümüzde pek çok yerel yer adları dizini bulunmaktadır. Ancak, bu dizinlerin her birinde yukarıda belirtilen farklı tanımlama düzeyindeki şemaların kullanılması, birlikte çalışabilirliği (interoperability) engelleyen en önemli faktördür [2]. Birlikte çalışabilirlik, yerel dizinlerin jenerik ve genel bir şema ile temsil edilmesi ile sağlanabilir. Böylelikle, bir yer adları dizininde tanımlanmamış olan coğrafi detay ve özneliliklerine, aynı şemayı kullanan diğer dizinlerden tek bir sorgu ile erişim sağlanabilir. Hill'in (2006) de belirttiği gibi, *"yerel dizinlerin hazırlanmasında varolan bir şemanın uyarlanması, yeni bir şema geliştirilmesinden daha iyidir"* [4]. Uzlaşılmış veya kabul görmüş ontolojiler veya ilişkisel kavramlar dizinleri sınıflandırmalarının kullanılması dizinlerin birlikte çalışabilirliğini kolaylaştırarak sorgulama sonuçlarını iyileştirir. Çizelge 2.2'de yer adları dizinlerine ilişkin şema veya KOS sunan web tabanlı dizinleri listelemektedir [4].

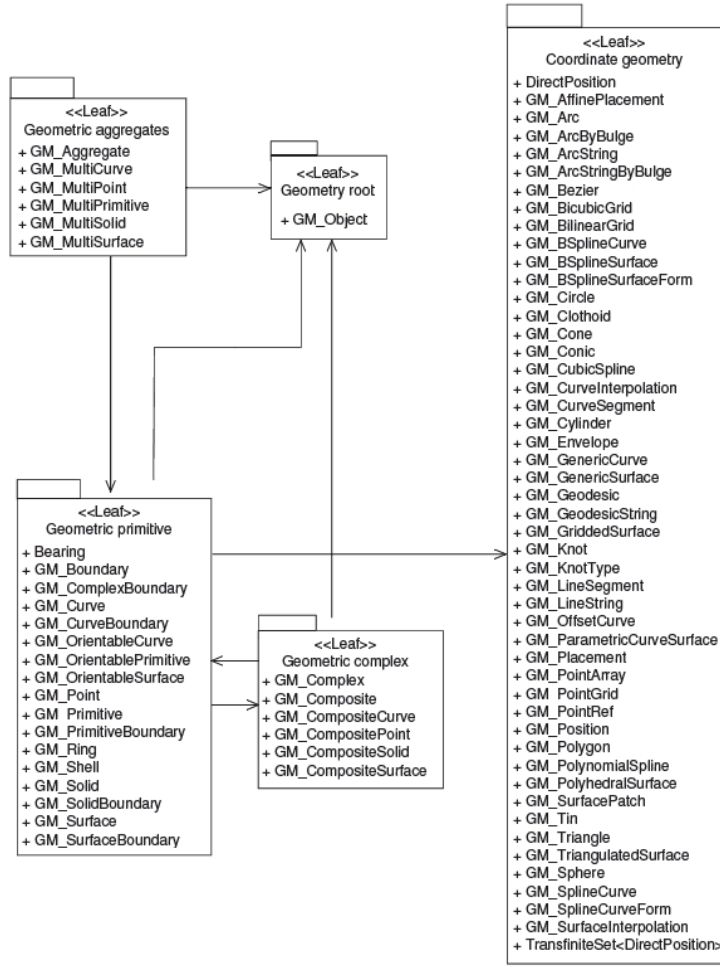
Çizelge 2.2 Web'deki yer isimleri dizinlerine ilişkin şemalar [4]

Organizasyon	Coğrafi detay tip tanımlamaları
U.S. Geological Survey	Alfabetik liste, 65 kategori ve tanımlamaları
U.S. National Geospatial-Intelligence Agency	9 sınıf, 643 kategori, kod isimleri ve tanımlamaları
National Resources of Canada	2 sınıf, 38 kategori, kodları ve terimleriyle
Geoscience Australia	117 kategorinin alfabetik listesi
Getty Thesaurus of Geographic Names	595 kategorinin alfabetik listesi
Alexandria Digital Library (ADL)	210 terim, 1046 tercih edilmemiş terim ve 6 üst terimle hiyerarşik kavramsal dizin

2.2.3 Konum

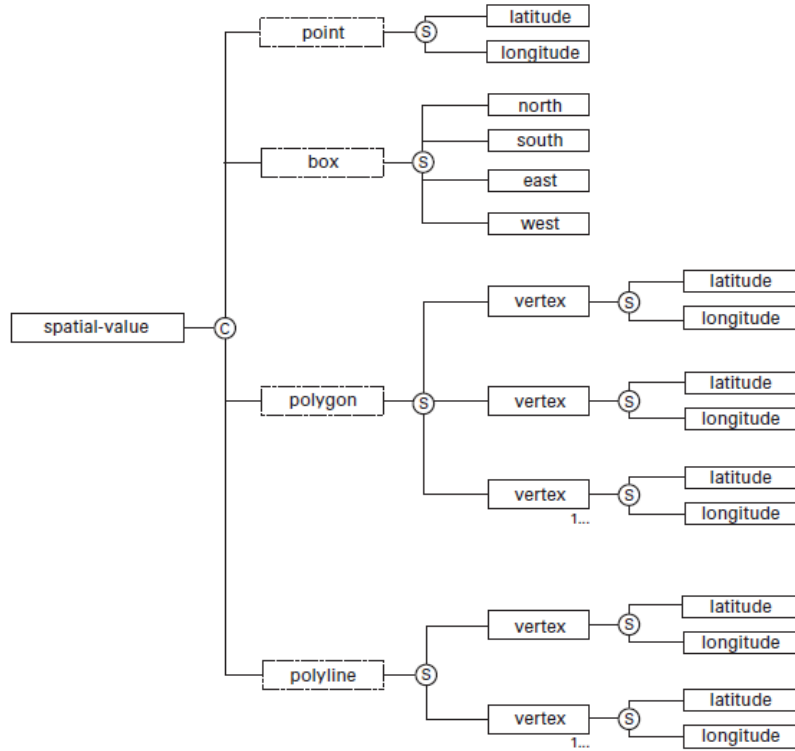
Konum, dünya üzerindeki adlandırılmış detay geometrisinin coğrafi koordinatlar veya diğer konumsal referanslar ile sunumudur [4]. Bir yerin konumsal biçimi farklı geometriler ile temsil edilebilir. Bu geometriler, Açık Konumsal Bilgi Konsorsiyumu (Open Geospatial Consortium – OGC) tarafından; nokta, kutu (box), düz çizgi (linestring), kapalı çizgi (linearring), poligon, çoklu nokta, çoklu düz çizgi, çoklu poligon ve çoklu geometri olarak belirlenmiştir [39]. Benzer şekilde, Uluslararası Standartlar Organizasyonu (International Organization for Standardization – ISO) 19107:2003 Coğrafi Bilgi – Konumsal Şema (Geographic information – Spatial schema) da, geometri ve topoloji için bir kavramsal veri teması sunar [40]. Şekil 2.3'te ISO tarafından kabul edilen geometri sınıfının özellikleri görülmektedir. Burada belirlenen geometri türleri, Coğrafi İşaretleme Dili (Geography Markup Language – GML) ile ifade edilebilir.

Dizinlerde konum genellikle nokta geometrisi ile temsil edilmektedir [4]. Ancak nokta geometrisi tüm coğrafi detayları temsil etmek için uygun değildir. Bazı coğrafi detayların poligon veya çizgi geometrileri ile temsili daha uygundur [36]. Örneğin bir il, merkezini gösteren nokta yerine sınırlarını gösteren poligonlarla temsil edilmelidir. Ancak, Hill [2006] tarafından belirtildiği gibi, detay geometrisinin kutu ya da poligonla temsili teknik ve maliyetli bir işlemdir [4]. Şekil 2.4, ADL Yer Adları Dizininde kullanılan geometri türlerini göstermektedir [41].



Şekil 2.3 ISO 19107 Geometri paketi sınıf ve özellikleri [40]

Coğrafi detay geometrileri, konumu temsil etmenin yanında, konumsal ilişkilerin kurulmasını da sağlar. OGC (1999) tarafından yayınlanan dokuz farklı topolojik ilişki ile coğrafi nesneler arasındaki bağlanabilirlik derecesi temsil edilir. Bu topolojik ilişkiler: (1) Eşittir (equals), (2) Ayrıktır (disjoint), (3) Kesişir (intersects), (4) Dokunur (touches), (5) Keser (crosses), (6) İçinde (within), (7) Kapsar (contains), (8) Kısmen örtüşür (overlaps) ve (9) İlişkilidir (relate) şeklindedir [42]. Bir yerin, her biri farklı zaman periyotları ile ilişkili olan, birden çok konumu olabilir [5]. Bu durumda, farklı zaman aralıklarına sahip coğrafi detayların geometrilerin ve topolojilerinin yeniden oluşturulması gerekebilir. Bölüm 3.9’da konumsal ilişkileri modelleyen çalışmalardan ve konumsal ilişkiler ifade eden sözlüklerinden, ontolojilerden bahsedilecektir.



Şekil 2.4 ADL konumsal veri sınıfları yapısı [41]

2.2.4 Yer Adları Dizinlerinin İlave Bileşenleri

Yer adları dizinleri, yukarıda açıklanan yer adı, detay sınıfı ve konum bileşenlerinin yanında diğer ilave bileşenleri veya öznitelikleri de içerebilir. EDINA (2007) ve Hill (2006) benzer biçimde bu ilave bileşenleri şöyle sıralamaktadır: (1) Alternatif adlar, (2) Konumsal ilişkiler, (3) Zamansal faktörler ve (4) Referans kaynakları [4], [6].

İlave bileşenlerden ilki, yerin alternatif adlarıdır. Bir yer birden fazla adla belirtiliyor olabilir. Örneğin, idari birimleri kapsayan bir dizinde idari birimlerin resmi adı kullanılır. Ancak, o idari birimin günlük hayatta kullanılan adı resmi adından farklı olabilmektedir. Alternatif adların dizinlerde temsili; dizinlerin sorgulama yeteneğini geliştirir ve ad değişikliklerinin yol açtığı karışıklıkları önler.

İkinci ilave bileşen, detayların uygun olduğu durumlarda nokta geometrisi yerine diğer geometrilerle temsili öngörür. Bazı detayların çoklu geometriler ile temsili konumun daha doğru ifade edilmesini sağlar, bununla beraber, girdiler arasındaki konumsal ve hiyerarşik ilişkileri güçlendirerek sorgulama sonuçlarını iyileştirir. Bu bileşen ayrıca, detaylar arasındaki topolojik ve mereolojik / partonomik ilişkilerin kurulmasını sağlar.

Topolojik ilişkiler, detaylar arasında binme ve çakışma gibi sorunları denetleme olanağı sağlar. Mereolojik / partonomik ilişkiler ise iki detay arasındaki parça - bütün ilişkisini ifade eder (örn., il, ilçe ve köy arasındaki ilişkiler). Bu tür ilişkilerin tanımlanması, bir detayın alt ya da üst hiyerarşideki ilgili diğer konumsal detaylara erişimi kolaylaştırır.

Üçüncü ilave bileşen zamansal faktörlerdir. Konum, konumlar arasındaki ilişkiler ve yer adı, zamanla değişebilir [4]. Dizinlerde zamansal özniteliklerin bulunması, güncelliği sağlar. Zaman özellikle, tarihsel yerler ve idari birimler gibi detaylar açısından önemlidir. Örneğin, idari birim adları ve bu birimlerin sorumluluk sınırlarının sıklıkla değişmesi, zaman bileşenin dizinlerde yer almasını zorunlu kılar. ADL Yer Adları Dizininde zaman; eski (former), güncel (current) ve önerilen (proposed) olarak üç öznitelikle ifade edilmiştir.

Son ilave bileşen, detayla ilgili referans kaynakların dizinlerde yer almasını öngörür. Bu, farklı kaynaklardan alınan bilgilerin (örn., yer isimlerinin, geometrilerinin, sınıflarının), kaynak bilgilerinin belirtilmesi anlamına gelir. Bir köyün veya ilçenin adının veya sınırlarının değiştirilmesine ilişkin bir idari karar; bu tür kaynaklara örnek gösterilebilir.

Yukarıda belirtilen ana ve ilave bileşenlerinin yanı sıra, yer adları dizinlerinde girdilere ilişkin meta veriler de kaydedilebilir. Örneğin, dizinin kapsamı, dizinde kullanılan verilerin doğruluğu, dizinin sahibi, güncellenme sıklığı, girdi sayısı, kullanma şartları vd. [4].

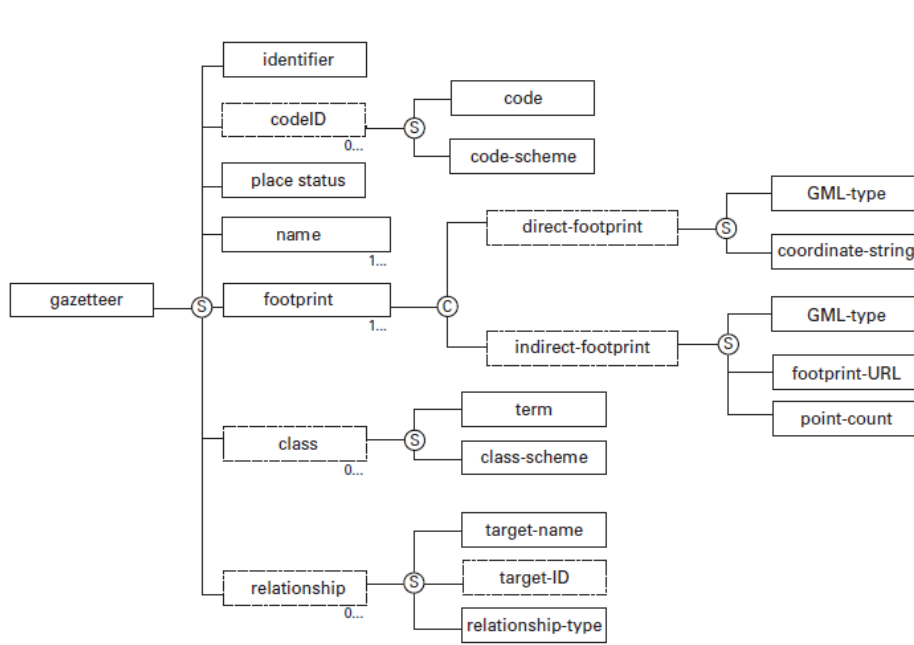
2.3 Küresel Yer Adları Dizinlerine İlişkin Standart Modeller

2.3.1 ADL Yer Adları Dizini Modeli

ADL, yer adlarına ilişkin küresel bir dizindir. Girişim kapsamında; bir yer adları dizini modeli (özet model ve içerik modeli), bu modeli yayınlamada kullanılmak üzere bir protokol ve detay sınıflarını belirlemek için oluşturulan bir ilişkisel kavramlar dizini geliştirilmiştir.

ADL, bir yer adları dizininin yarı-formel modelini içeren 'ADL Yer Adları Dizini Özet Modeli'ni (abstract model) içermektedir (Şekil 2.5). Şema, dizin girdileri için seçimlik elemanları kesik çizgili kutularla, zorunlu elemanları sürekli çizgili kutularla tanımlar.

Buna göre, bir yer adları dizini girdisi, bir tanımlayıcı (identifier) ile ifade edilir [43]. Bunun yanında 1 girdi; 1 veya daha fazla yer adı, 1 veya daha fazla konum, 0 veya daha fazla detay sınıfı bilgisi içermelidir [4], [43]. Bir girdi için birden çok ad, konum ve detay sınıfı varsa birincil bir ad (öncelikli ad), geometri ve detay sınıfı belirlenir [4]. Kullanılan üç zorunlu bileşen ve tanımlayıcı beraber eşsiz (unique) bir girdi oluşturmaktadır. Bunun yanında özet modelde kod, konum, detay sınıfı ve ilişkiler için farklı alternatifler olduğu görülmektedir.



Şekil 2.5 ADL Yer isimleri dizini özet modeli [4]

ADL, yukarıda açıklanan özet modelin yanı sıra, Yer Adları Dizini İçerik Standardı (ADL Gazetteer Content Standard – GCS) başlıklı bir standardı kullanmaktadır. GCS, özet modelinde belirtilen bileşenlerin yanında, her girdinin içeriğini tanımlamıştır. Yer adları dizini özet modeli Genişletilebilir İşaretleme Dili Şema Tanımlaması (Extensible Markup Language Schema Definition – XSD) olarak yapılandırılmış; hem şema olarak hem de ilişkisel veritabanı olarak online yayınlanmıştır [43]. GCS'nin, en üst düzey yapısı Şekil 2.7'da gösterilmektedir:

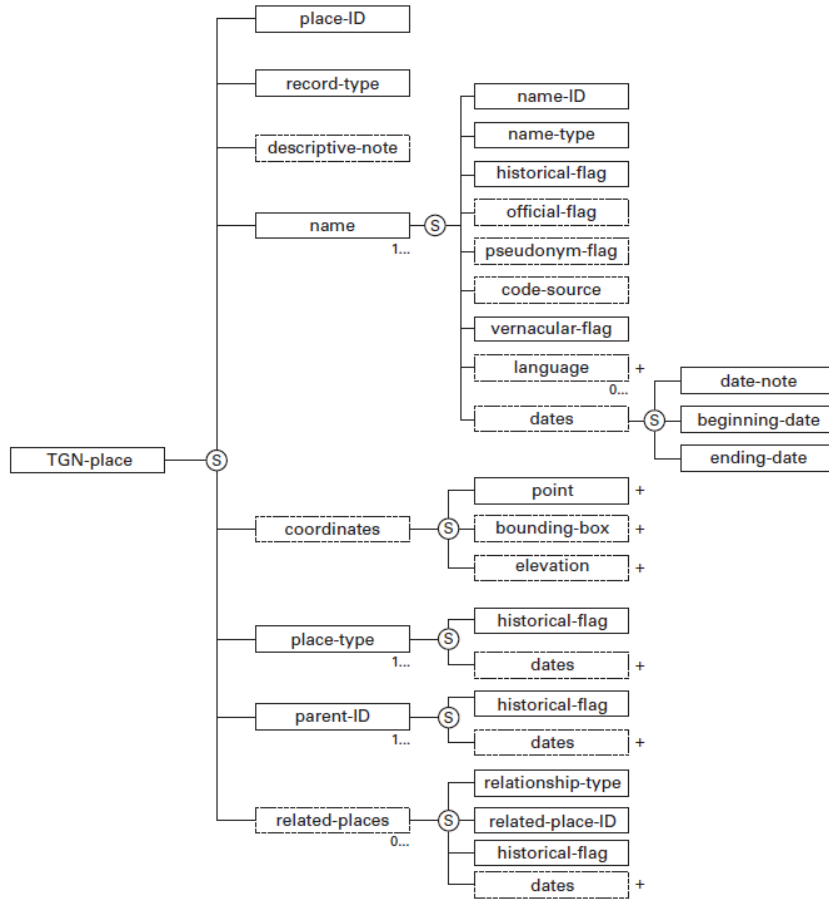
- 1) Her bir dizin girdisi için eşsiz bir coğrafi detay tanımlaması (ID),
- 2) Coğrafi detayın zamansal durumu (eski, güncel, önerilen),

- 3) Coğrafi detay için en az bir ad ve bu adın zamansal durumu,
- 4) En az bir sınıf terimi veya sınıflandırma numarası, kullanılan şemasının tanımlaması ve sınıfın zamansal durumu,
- 5) Geometri veya kutu (bounding box) olarak ifade edilmiş en az bir konum (footprint) ve bu konumun zamansal durumu,
- 6) İdari (yönetimsel) veriler: Girdinin oluşturulma ve güncellenme tarihi.

Detay sınıfları, ADL Coğrafi Detay Tipi İlişkisel Kavramlar Dizinini'ne (ADL Feature Type Thesaurus – FTT) göre tanımlanır. FTT coğrafi yerleri sınıflandırmak, detay tiplerine ayırmak için bir dizi terim ve sınıftan oluşturulmuş hiyerarşik bir sınıflandırma listesidir [43]. FTT; idari alanlar (administrative areas), hidrografik alanlar (hydrographic areas), arazi parselleri (land parcels), insan yapımı coğrafi detaylar (manmade features), fizyo-coğrafik detaylar (physiographic features) ve bölgeler (regions) olmak üzere 6 ana terim altında gruplandırılmış 210 terim (tercih edilen) ile bu terimlerin 1046 alternatifini (tercih edilmeyen) içerir.

2.3.2 Getty Coğrafi Adlar İlişkisel Kavramlar Dizini

Getty TGN, yer adları için bir ilişkisel kavramlar dizini modelidir. 2.035.195 adet yer adı ve bunların konumlarını içeren TGN, ISO standartlarına uygun olarak yapılandırılmıştır. Şekil 2.6'da TGN dizininin genel yapısı görülmektedir. Her dizin girdisinin bir adı, bir konum tipi ve bir konum bilgisi olması gerekmektedir. TGN dizininde geometriler nokta veya kutu olarak ifade edilebilmektedir. Bunun yanında her adın eşsiz bir tanımlayıcısı bulunmaktadır. Adların zamansal değişimleri başlangıç ve bitiş tarihleri ile sağlanmaya çalışılmıştır. Her dizin girdisinin ilişkili olduğu yerler belirtilebilmektedir. İncelenen diğer dizinlerden farklı olarak TGN, yükseklik bilgisi de sunmaktadır [47]. ADL Yer Adları Dizininden sonra en kapsamlı klasik yer adı dizini TGN'dir.

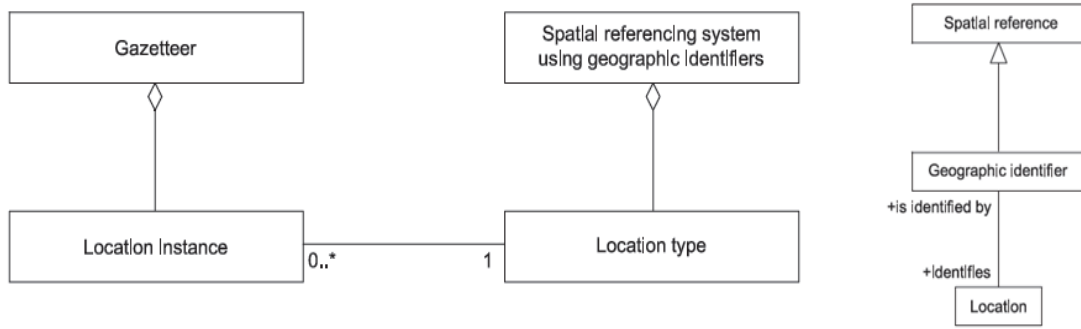


Şekil 2.6 Getty TGN modeli [4]

2.3.3 ISO 19112 Yer Adları Dizini Standardı

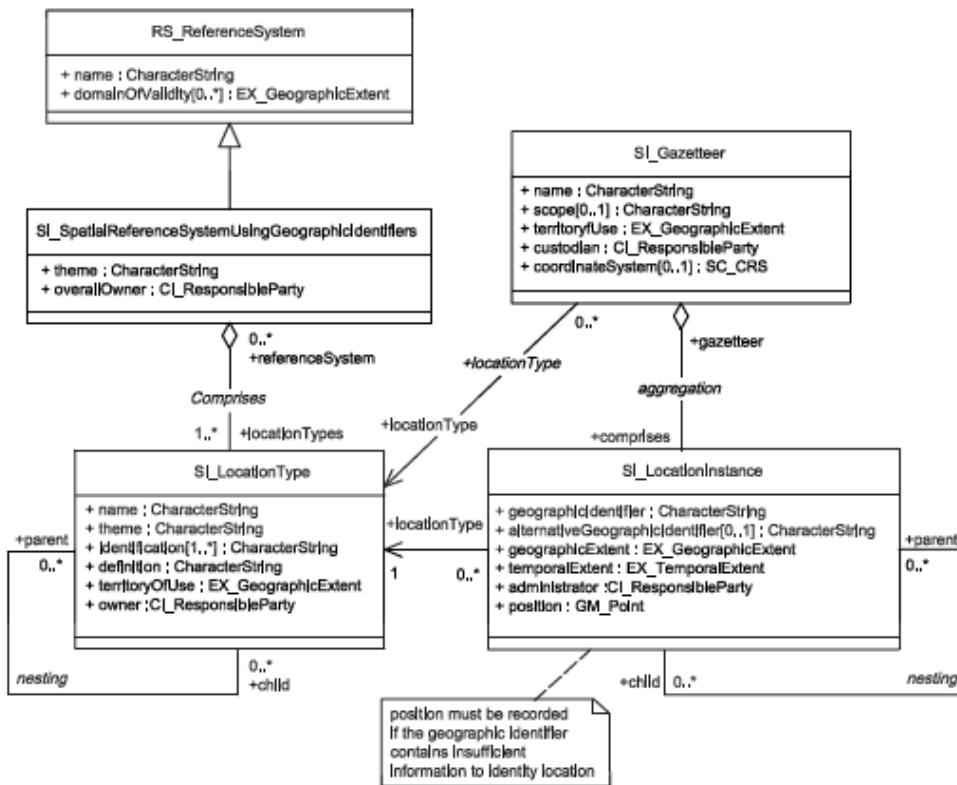
ISO 19112:2005 Coğrafi Bilgi – Coğrafi Tanımlayıcılarla Konumsal Referanslama (Geographic Information—Spatial Referencing by Geographic Identifiers); coğrafi tanımlayıcılarla konumsal referanslama sağlayan uluslararası bir yer adları dizini modelidir.

Şekil 2.7’de görüldüğü gibi, standartta konum bir coğrafi tanımlayıcı ile tanımlanır ve bu tanımlayıcı konumsal referans olarak kullanılır. Standart, yer adları dizinini coğrafi tanımlayıcılar (örn., ülke adı, ülke kodu gibi) tarafından tanımlanmış konum örneklemelerinin (instance) bir altdizini olarak tanımlamıştır. Konum örneklemeleri (location instance) yer adları dizinini, konum tipleri (location type) ise coğrafi tanımlayıcıları kullanan konumsal referans sistemlerini oluşturmaktadır. Bütünleşik Modelleme Dilindeki (Unified Modeling Language – UML) genişletilmiş model mimarisi Şekil 2.8’de görülmektedir [3].



Şekil 2.7 ISO 19112:2005 Coğrafi tanımlayıcılarla konumsal referanslama yapısı [3]

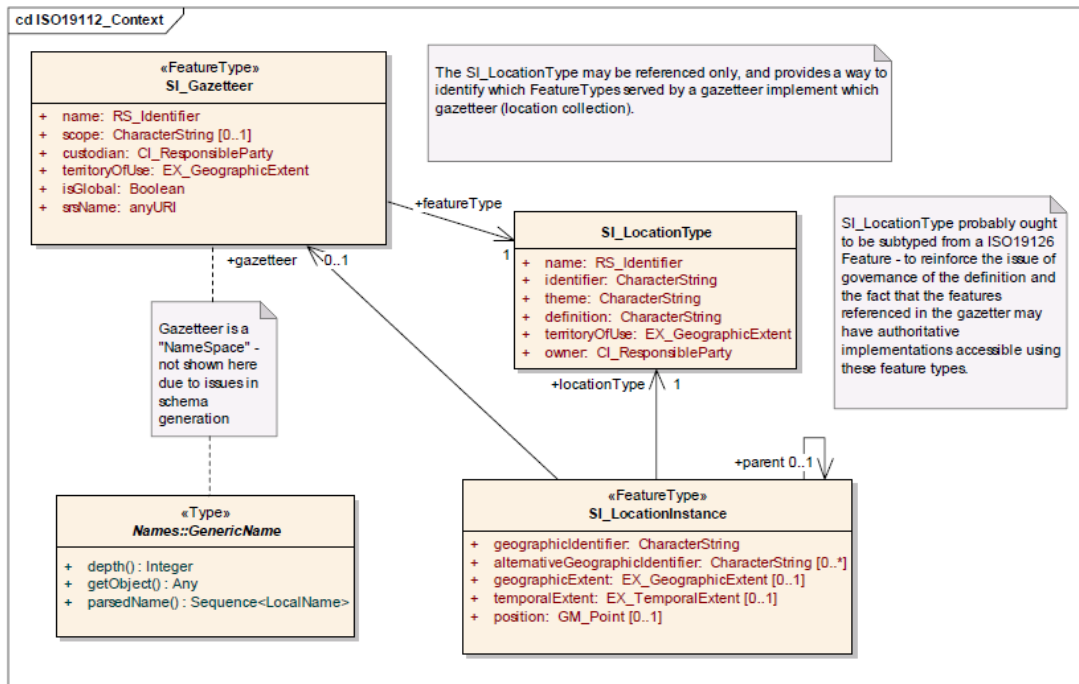
Genişletilmiş modelde, yer adları dizini sınıfının konum örneklemeleri sınıfından oluştuğu görülmektedir. Her konum örnekleme bir eşsiz tanımlayıcısı vardır. Ayrıca alternatif bir tanımlayıcısı bulunabilir. Konum örneklemelerinin geometrisinin nokta olarak ifade edileceği belirtilmiştir. Her konum örnekleme bir konum tipi olması gerektiği ‘Konum Örnekleme’ sınıfı ile ‘Konum Tipi’ sınıfının arasında kurulmuş ilişki ile belirtilmiştir. Konum tipleri birleşerek konumsal referans sistemlerini oluşturmaktadır [3].



Şekil 2.8 ISO 19112:2005 UML modeli [3]

ISO 19112 standardı modelinde yer adlarının eşsiz biçimde dizinlerde tanımlanması gerektiği belirtilmiştir. Dizin girdileri için eşsiz birer tanımlayıcı alan bu adlar öncelikli ad olarak ifade edilmiştir. Standartta bir adın, birden fazla yer adları dizini girdisinde kullanılabileceği; fakat birden fazla girdide öncelikli ad olarak kullanılamayacağı belirtilmiştir. Eski yer adları dizini yaklaşımlarında, sınıflandırma bilgileri dizin modelinden ayrı, bir KOS olarak sunulmaktaydı. Önceki yer adları dizinlerinden farklı olarak ISO, sınıflandırma bilgilerini dizinin bir parçası olarak modellemiştir.

Şekil 2.9'de ISO 19112:2005 veri modelinin, GML formatında uygulama şemalarına temel olan ISO 19118'e göre yeniden şekillendirilmiş (refactored) yapısı görülmektedir [44]. Modelin mantıksal yapısında bir değişiklik yapılmamıştır, sadece yeniden yapılandırılmıştır [45], [46].



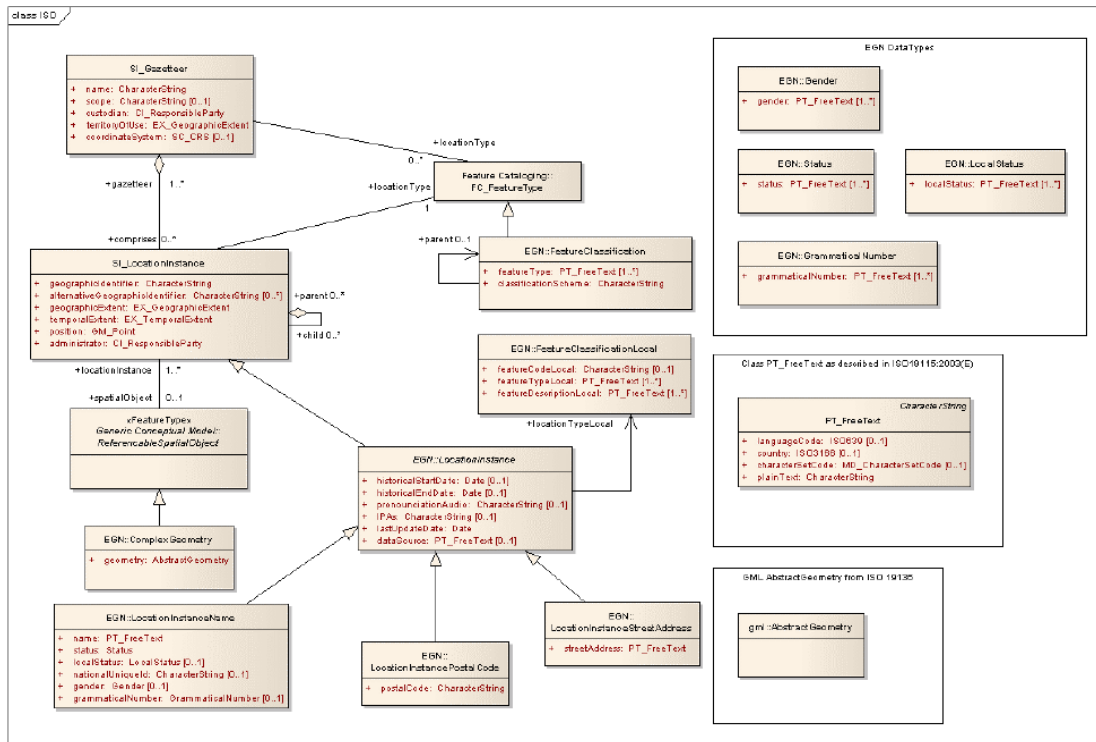
Şekil 2.9 Yeniden şekillendirilmiş ISO 19112 [46]

2.3.4 Avrupa Coğrafi Adlar Dizini

Avrupa Coğrafi Adlar Dizini (EuroGeoNames – EGN), Avrupa veri altyapısı için coğrafi adlardan yararlanma ve bu adları yönetme amacıyla geliştirilmiş bir projedir. Ulusal Harita ve Kadastro Kurumlarından (National Mapping and Cadastral Agencies - NMCA) alınan coğrafi bilgileri düzenleme ve servis etme amacını taşır. [6], [48]. EGN projesi

kapsamında 2008 yılına kadar olan tüm yer adları dizinleri incelenmiş ve en uygun yapı bulunmaya çalışılmıştır. EGN, bu incelemeler sonucunda, geliştireceği yer adları dizini modelinde ağırlıklı olarak ISO 19112:2005 yer adları dizini veri temasından ve INSPIRE Coğrafi Adlar Veri Temasından (2007) yararlanma kararı almıştır.

Şekil 2.10’da görülen EGN kavramsal modeli, ISO 19112:2005 yer adları dizini standardının ana yapısı üzerine kurulmuştur. ISO 19112 ‘Yer Adları Dizini’ ve ‘Konum Örneklemeleri’ sınıfları, hiçbir değişiklik yapılmadan EGN modelinde kullanılmıştır. INSPIRE jenerik kavramsal model sınıflarından ‘Referanslandırılabilir Konumsal Nesne’ ve ‘Coğrafi Detay Tipi’ sınıfları birkaç öz nitelikle genişletilerek kullanılmıştır. EGN’de yer adlarının, posta kodu ve adres verilerinin de öz nitelik olarak ifade edilmesi gerektiği belirtilmiştir. EGN, yer adları dizin protokolü olarak bir Basit Nesne Erişim Protokolü (Simple Object Access Protocol – SOAP) uygulaması olan, OGC’nin Web Harita Saklama Servisi’nin (Web Map Tile Service – WMTS) kullanılmasını önermektedir [6], [51], [52].



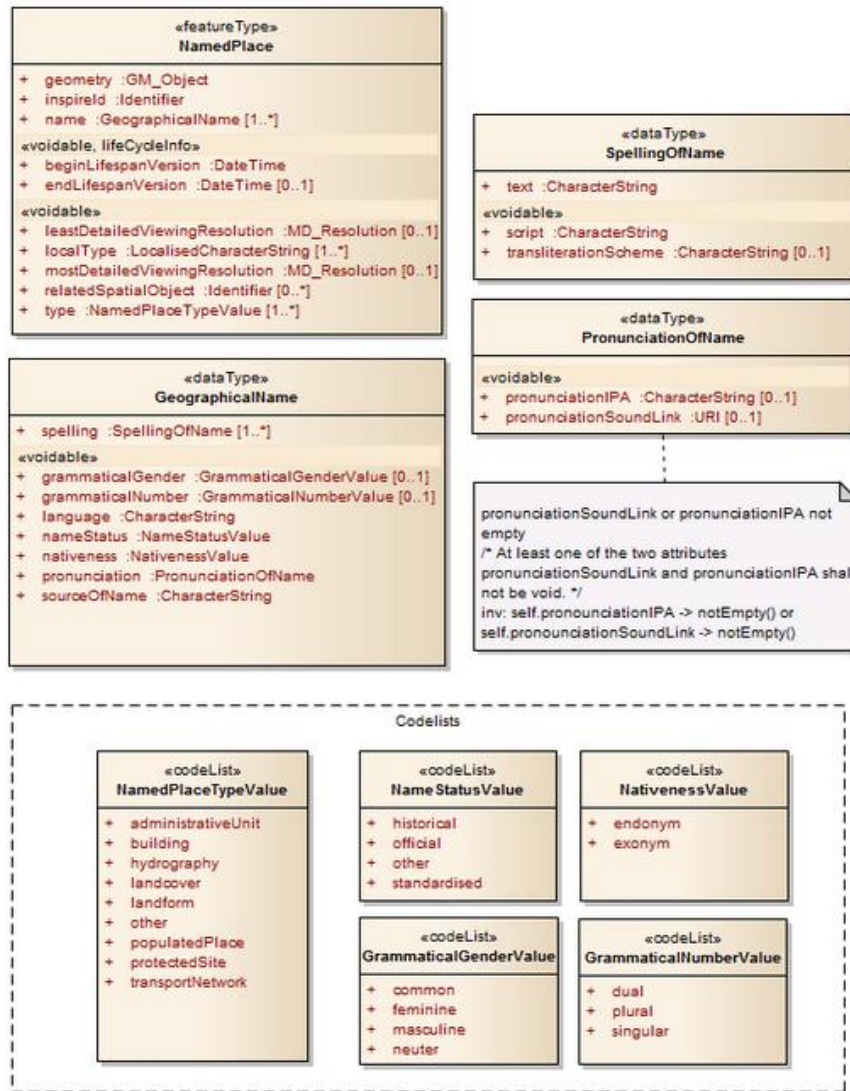
Şekil 2.10 EGN kavramsal veri şeması [52]

2.3.5 INSPIRE Coğrafi Adlar Standardı

INSPIRE tüm Avrupa’da Konumsal Veri Altyapısını (KVA) oluşturmak, konumsal verileri erişilebilir ve birlikte çalışabilir kılmayı amaçlayan bir projedir. INSPIRE kapsamında coğrafi verilerin (örn., idari birim, kadastral parsel, bina vd.) temsiline ilişkin standart veri tanımlamaları yapılmıştır. Bu veri tanımlamaları içinde diğer veri temalarına da altık oluşturan Coğrafi Adlar Veri Teması, coğrafi yer adları dizinlerinin standardizasyonunu amaçlar. Coğrafi yer adlarının modellendiği en güncel veri modeli olan INSPIRE Coğrafi Adlar Veri Teması, uluslararası anlamda yer adlarını ifade eden en kapsamlı veri modelidir. INSPIRE da, ISO’ya benzer şekilde, verilerin yayımlanmasında Servis Yönelimli Mimari’yi (Service Oriented Architecture - SOA) önermektedir.

INSPIRE Coğrafi Adlar Veri Uygulama Şeması, Şekil 2.11’de gösterilmektedir. ‘Adlandırılmış Yer’ (NamedPlace) sınıfı; eşsiz bir INSPIRE tanımlayıcısı ile tanımlanan ve bir geometrisi olan gerçek dünya objesinin bir veya birden fazla adla tanımlanmasını sağlayacak şekilde modellenmiştir. Diğer yer adları dizinlerinden ve modellerinden farklı olarak, geometri tipi olarak poligon ve çoklu poligon dahil birçok geometrinin kullanılabileceği, ‘Adlandırılmış Yer’ sınıfında ifade edilmiştir. ‘Adlandırılmış Yer’ sınıfında ifade edilen her coğrafi adın, ISO 19112 standardında olduğu gibi, en az bir sınıfı olması gerekmektedir. INSPIRE standardına göre, her ada eşsiz bir tanımlayıcı atanmalıdır. Adlar eşsiz olmak zorunda değildir. Zamansal faktörler dizin girdilerinin başlangıç ve bitiş tarihleri ile kaydedilir. Modelde, dizinle ilgili meta verilerin saklanması önerilmiş, ama zorunlu tutulmamıştır [7].

‘Coğrafi Ad’ sınıfı, ‘Ad Telaffuzu’ ve ‘Ad İmlası’ (SpellingOfName) sınıfları ve coğrafi adların kaynağı/dili vd. öznitelikleri ile aynı coğrafi adın farklı telaffuzları ve yazımlarına ilişkin yapıları ifade eden kod listeleri, bir bütün olarak tanımlanmıştır. Örneğin, ‘Adlandırılmış Yer’ sınıfına ait “Athina” adlı bir objenin adının farklı telaffuzu olan “Athena” ‘Ad Telaffuzu’; farklı yazımı olan “Αθήνα” ‘Ad İmlası’ sınıfları ile tanımlanır [7].



Şekil 2.11 INSPIRE coğrafi isimler veri teması [7]

2.4 Uluslararası Yer Adları Dizin Protokolleri

Günümüzde sayısal yer adları dizinleri, kaynaklarına erişim için Zengin Metin İşaret Dili'nin (Hyper-Text Markup Language – HTML) kullanıcı arayüzü veya Web CBS yazılımları kullanmaktadır. Birbirinden ayrı ve bağımsız yer adları dizinleri, servis protokolleri sayesinde birleşik biçimde sorgulanırlar. Dağıtık biçimde bulunan yer adları dizinlerinden bütünleşik sonuçlar alınmasına olanak verirler. Bu protokoller, sorgulamaların dizinlere gönderilme ve kullanıcının dizinlerden cevap alma aşamalarını formüle eder. Servis protokolleri bir yer adı sorgusunu konumsal sorguya çevirme; harita, uydu fotoğrafı gibi bir altlık üzerinde yerin konumunun gösterilmesini sağlar [4].

2.4.1 ADL Yer Adları Dizini Protokolü

ADL yer adları dizini protokolü (ADL Gazetteer Protocol), yer adları dizinlerine erişim için tasarlanmış XML ve HTML tabanlı bir protokoldür. Bu açık kaynak kodlu dizin protokolü, Java dilinde yazılmış istemci ve sunucudan oluşmaktadır. İstemci, sorgulama ve sorgu sonuçlarını altlık bir harita üzerinde gösterme özelliklerine sahiptir. Sunucu ise yer adları dizini veri tabanını, protokol operasyonları ile eşleştirir [49].

ADL protokolü yer adları dizinlerini; tanımlayıcılar, zamansal bilgiler (güncel, tarihsel, önerilen), yer adları, konumlar ve coğrafi detay sınıfları ile yapılan sorgulamaları destekler. Protokol, diğer servislerle bütünleşik sorgulamalara olanak sağlamak için tanımlamalarını çok basit tutmuştur. Protokol, her sorguda farklı operasyon kullanır. Örneğin, konum sorgulamalarında; içerir ve içindedir operatörleri kullanılır. Sorgulama yanıtları, protokol tarafından belirlenen standart rapor, XML formatında sunulur [4], [49].

2.4.2 OGC Yer Adları Dizini Protokolü

OGC yer adları dizini mimarisi, bir portal ve bu portalın yer isimleri dizini alt servislerinden oluşur [5]. 2004 yılında yayımlanan Konumsal Portal Referans Mimarisi (Geospatial Portal Reference Architecture), 2006 yılında yayımlanan Yer Adları Dizini Web Coğrafi Detay Servisi Uygulama Profili (Gazetteer Service - Application Profile of the Web Feature Service Implementation Specification, versiyon 0.9.3) ve 2012 yılında yayımlanmış (Gazetteer Service - Application Profile of the Web Feature Service Best Practice) en temel tanımlamalardır [45], [46], [50].

OGC portalı, birçok kaynağa bir erişim noktası sağlayan bir geçit olarak tanımlamıştır [50]. Portallara iletişimin temel metodu servis değişim paradigmasını (servis trade paradigm) izleyen Servis Yönelimli Mimariye dayanır. Portalların kullanım amacı, Web üzerinden çeşitli konumsal verilerin paylaşımı, değişimi ve birlikte çalışabilirliğini sağlamaktır. OGC tarafından oluşturulan protokolleri, şemaları, arayüzleri ve kodlamaları kullanarak, uluslararası standartlara uygun başka portallar oluşturulabilir.

Konumsal Portal Referans Mimarisi, SOA yapısına uygun olarak tasarlanmıştır. SOA, yazılım fonksiyonelliği sunan, kolayca genişletilebilen, entegrasyonu destekleyen,

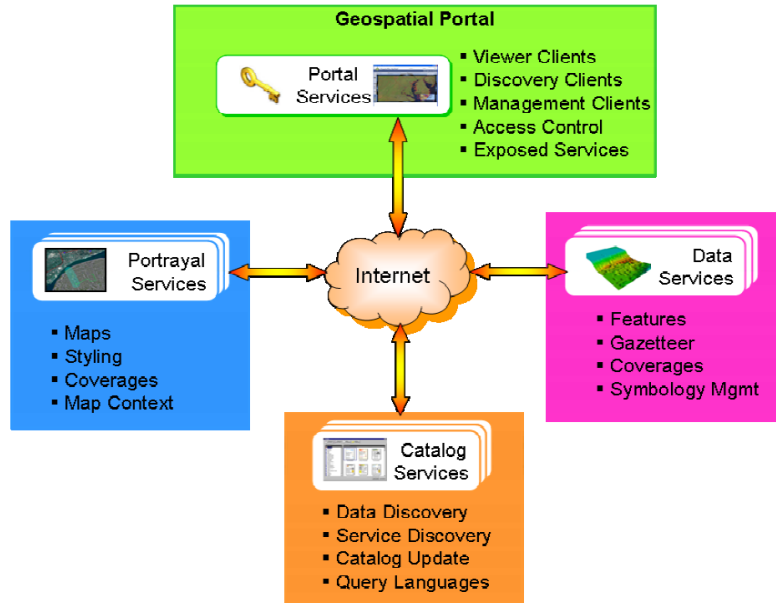
esnek ve açık kaynak kodlu bir yapıdır [50]. SOA, Web servisleri üzerine temellendirilmiştir. Web servisleri, yazılımlara standart arayüzler sağlar. Servis sahipleri, bir servis kaydı (registry) oluşturarak bilgilerini paylaşabilir. Servis kullanıcıları da, paylaşılan servisleri araştırabilir, servis bilgilerine erişebilir.

Bir konumsal portalda bulunması gereken temel dört servis Şekil 2.12'deki gibi OGC tarafından şöyle belirlenmiştir: **(1)** Konumsal bilgiye portal üzerinden tek bir noktada erişim sağlayan portal servisi (Portal Services); **(2)** Konumsal servisleri ve veri setlerini bulmak için kullanılan katalog servisleri (Catalog Services); **(3)** Konumsal bilgileri işlemek ve kullanıcıya sunmak için kullanılan tanımlama servisleri (Portrayal Services) ile **(4)** Konumsal içerik ve verinin işlenmesi sağlamak için kullanılan veri servisleri (Data Services). Yer adları dizini alt servisleri ise kullanıcıya; **(1)** coğrafi detayları araştırma olanağı veren dizin istemcisi (gazetteer client), **(2)** yer adlarını sorgulama olanağı sağlayan dizin servisi (gazetteer service) ve **(3)** dizin içerik modelinden (gazetteer content model) oluşmaktadır.

Dizin istemcisi coğrafi detayları, coğrafi adlar aracılığıyla aramaya olanak sağlar. Coğrafi detaylara erişim için sorguları formüle etmeye izin verir. Tanımlama servisleri genel olarak, Web Harita Servisi (Web Map Service – WMS) ve Kartografik Düzenlemeler Servisi (Styled Layer Descriptor – SLD) servislerini kullanarak istemciye çeşitli formatlarda (tiff, gif, png, jpeg) çıktı sunma amacına hizmet etmektedir.

Veri servisleri 'Feature Services', 'Symbolology Management', 'Gazetteer', 'Coverage Services' bileşenlerinden oluşmaktadır. Burada 'Feature Service', WMS tarafından sağlanan sorgulama özelliği kullanılarak coğrafi detayları sorgular ve Zengin Metin Transfer Protokolü (Hyper-Text Transfer Protocol – HTTP) istemcisine GML olarak cevap verir. Yer adları dizini servisi bir sorgulama isteği ile bir veya daha fazla coğrafi detaya erişim sağlar. Coğrafi detaylar; öznitelikleri, sınıfları, adları ve tanımlayıcıları (id) aracılığıyla sorgulanabilir. OGC yer adları dizini servisi, WFS'nin özelleşmiş bir şeklini (WFS-G) XML/HTML formatında geliştirmiştir. Burada amaç, OGC tarafından geliştirilen diğer standartların, WFS-G tarafından kullanılabilmesi, servislerin birlikte çalışabilmesi, diğer standartların yeteneklerinin kullanılabilmesidir. Sözü edilen servis ve alt servis yapısının kullanılması, heterojen coğrafi bilgi içeriğinin değişimini ve Web üzerinde

çeşitli konumsal servisler kullanılarak yayınlanan bilginin paylaşımını olanaklı kılar; birlikte çalışabilirliği sağlar.



Şekil 2.12 Konumsal Portal Referans Mimarisi Servis Dağılımı [50]

OGC yer adları dizini protokolü (gazetteer protocol) WFS'nin geniş konumsal sorgu operasyonlarını (kesişir, dokunur vd.) miraslayabilmesi ve GML'in yaygınlığından faydalanabilmesi dolayısıyla avantajlar sağlar [4]. Bir dizin servisinin sorgulamayı desteklemesi için, protokol tarafından bazı operasyonlar sağlanmaktadır:

- 1) '**getCapabilities**', hangi coğrafi detay tiplerinin servis edildiğini, hangi operasyonların coğrafi detay tipleri üzerinde uygulanabileceği ve yer adları veri tabanının nasıl oluştuğu hakkında bilgi veren, XML formatında bir dosya sunar. Servislerin tüm kabiliyetlerine ait bilgiler bu operasyon uygulanarak ortaya çıkarılabilir.
- 2) '**describeFeatureType**', dizin istemcisinin şema tanımlamalarına erişimini sağlar. Şema tanımlamaları, dizin sunucusunun coğrafi detay örnekleme çıktılarını nasıl oluşturulacağını anlatır. '**getFeature**' operasyonu yanıtlanarak bu operasyon yapılır. Bu operasyonun tanımı, bir ISO 19112 sınıfı olan 'Konum Örneklemeleri' temel sınıfında üretilir.

- 3) **'getFeature'**, yer adları dizininin coğrafi detay örneklemelerine erişim sağlar. Bu operasyon WFS-G tarafından yürütülür. GML formatlarında çıktı alınabilir. **'getFeature'** operasyonu, yer adları dizinindeki ve portaldaki her bir yer adı dizinindeki girdilere erişim sağlayabilmelidir.
- 4) **'getGMLObject'**, yer adı dizini örneklemelerine erişimi xLinks aracılığıyla sağlar [45].

OGC protokolü tarafından yer adları dizininde yapılan sorgu şu şekilde gelişmektedir: **(1)** İstemci uygulaması WFS-G'den **'getcapabilities'** ile servis kabiliyetlerini ister. **(2)** WFS-G tarafından servis edilen coğrafi detaylardan bir veya daha fazlası, istemci tarafından **'getFeature'** ile WFS aracılığıyla istenir. **(3)** **'Konum Örneklemeleri'** sınıfından coğrafi detaylar seçilir. **(4)** **'Konum Tipi'** sınıfı tanımlamalarına göre WFS arayüzünde bir istek oluşturulur. **(5)** WFS-G bu isteği okur ve servis eder. **(6)** WFS-G bu isteği tamamladığında bir rapor oluşturur ve istemciye gönderir. Burada sıklıkla bahsedilen WFS-G profilinin oluşturulmasındaki amaç, servis kayıtları (registries) aracılığıyla uygun WFS-G servislerini keşfetmek ve sunmaktır. WFS-G yer isimleri dizinlerine kolay ve anında ulaşım sağlar. OGC, ISO 19112 yer adları dizini standardının WFS ve WFS-G ile sunumunu önermektedir [45], [46].

SEMANTİK WEB ve BAĞLI VERİ

Tim Berners-Lee tarafından 1989 yılında HTML ve internet üzerinde çalışan Web ortamının geliştirilip açık bir şekilde toplumun hizmetine sunulmasından bu yana, Web alanında birçok değişiklik, yenilik yaşanmıştır. 2000’li yılların başından itibaren yaygın olarak kullanılan Web servisleri günümüzde; Web 2.0, Klasik Web, Geleneksel Web, Doküman Web’i gibi isimlerle anılmaktadır. Bu geleneksel yapıda, Web hiper bağlantılarla (hyperlink) birbirine bağlanan dokümanlardan oluşur. Web kullanıcıları, hiper bağlantının yönlendirdiği Web sayfasında aradığı bilgiye ulaşmayı umut eder, fakat arama motorlarının yönlendirdiği Web sayfasında ne olduğunu bilmez [17].

Geleneksel Web, dokümanlar dışındaki başka verilerin de Web uygulamaları ve servisleri olarak da sunulmasına olanak verir. Son dönemde birçok Web sitesi, sosyal paylaşım siteleri, kendi servislerinin yeteneklerine erişim ve uygulamalarının arkasındaki verilere Uygulama Programlama Arayüzü (Application Programming Interface – API) yardımıyla erişim sağlamaktadır. API’ler servislerin hangi sorgulamalara nasıl cevap vereceğini belirler. Genel Nesne İstek Aracı Mimarisi (Common Object Request Broker Architecture – CORBA), SOAP ve yakın dönemde Temsili Durum Transferi (Representational State Transfer – REST) Web servisleri olarak kullanılmaktadır. Bunlar içinde, REST mimarisi servislerin sağladığı kaynakları tanımlamak üzerine yoğunlaşır. Veri çağırma, silme ve sorgulamada HTTP isteklerini kullanır. Virgülle Ayrılmış Dosyalar (Comma-Separated Values – CSV), Javascript Nesne Değişkeni (Javascript Object Notation – JSON), XML gibi yarı yapılandırılmış ve yapılandırılmış veri formatlarında karışık veriler sunulabilir. REST servisleri veri erişim

mekanizmasını kolaylaştırmasına rağmen, Web servislerine özel olarak oluşturulan veri yapıları ve veri setleri arasında bağlantı kuramaz. Bunun yanında birçok API, varlıklarını lokal olarak tanımlar. API'lerdeki varlıkları tanımlayacak standart bir mekanizma yoktur. Bu nedenle, API'lerdeki veriler içeriğinden ayrılınca anlamsızlaşır. Bir programcı farklı API'leri kullanarak bir uygulama yazmak istediğinde, her API'den dönecek veri yapısını bilmeli, veri setleri arasındaki ilgili bağlantı noktalarını keşfedip birbirine bağlamadır. İyi kodlanmış bir API aracılığıyla bu işlem görece kolaydır, fakat otomatik olarak yapılamaz. Ayrıca API'lerde dokümanlardan arama motorlarına referans bağlantılar sunan HTML özellikleri 'anchor' ve 'href' tanımlanamadığından, dokümanlar arasında bağlantılara ulaşılamaz, veri setlerinin keşfi zorlaşır. Sonuç olarak, Web API'lerinin kullandığı veriler Web'de izole edilmiş parçalar olarak bulunmaktadır [14], [15], [17].

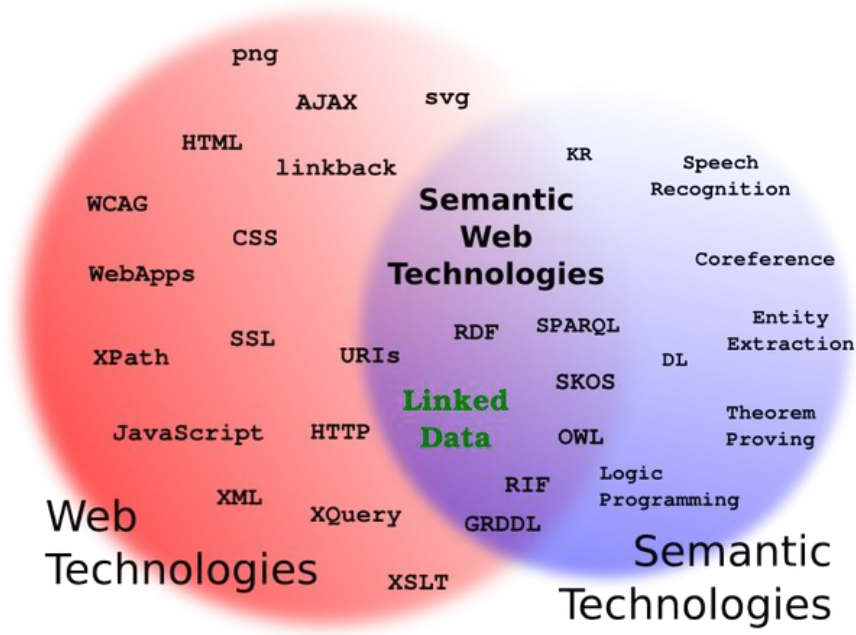
Web, veri paylaşımında oldukça etkin bir araçtır. Bu araçta; yeniden kullanım için veriye kolay erişim sağlama, veri setleri arasındaki ilişkilerin tanımlanması, kaynağı bilinmeyen verilerin uygulamalara entegre edilmesi ve arama sonuçlarını iyileştirmek için yeni bir yaklaşıma gereksinim duyulmuştur: Semantik Web [13], [14], [15], [17], [53]. 1990'ların son döneminde Sir Tim Berners-Lee tarafından öne sürülen Semantik Web kavramı ile sadece dokümanların birbirlerine bağlanabildiği, veri setleri arasında bağlantı sağlamayan Geleneksel Web'e yeni bir bakış açısı getirilmiştir. Semantik Web, sadece dokümanların değil, veri setlerinin de birbiriyle bağlanmasına olanak veren küresel bir bilgi uzayı olarak tasarlanmıştır [15]. Geleneksel Web'in merkezi yapısına karşın, Semantik Web merkezleştirilmiş (decentralized) yapıdadır. Semantik Web, veri setleri arasında anlamsal bağlantılar kurarak makinelerin ve insanların verileri anlamasını ve paylaşmasını sağlar [54].

Geleneksel Web ve Semantik Web arasındaki yapısal ve işlevsel fark, Bizer ve Heath (2011) tarafından şöyle örneklendirilmiştir: Geleneksel Web'de kitaplar ve kitapların dağıtıldığı kitapçılar için bir uygulama yapılmak istendiği zaman; kitaplar için tanımlanmış bir API, kitapçılar için bir API ve kitapçıların konumları ile ilgili ayrı bir API tasarlanmalıdır. Semantik Web'de ise RDF veri modeli kullanılarak bu üç farklı API birbiriyle bağlantılı hale getirilir, API'nin sağladığı veriler belirli bir formatta yayımlanabilir ve başka kullanıcıların veriye ulaşabilmesi ve yeniden kullanılması

sağlanabilir. RDF, API'lerin aksine sadece dokümanları değil, verileri de birbirine bağlama olanağı sağlar. Veri setleri arasındaki ilişkiler, bağlantılar Geleneksel Web'in aksine açıkça belirtilir. Bu bağlantılar sayesinde belirli bir alanla ilgili daha çok veri keşfi sağlanır, Web keşfedilebilir ve veri setleri uygulamalar tarafından kullanılabilir hale gelir [14]. Semantik Web, Geleneksel Web'de farklı servislerden sunulan farklı formattaki verilerin aynı anda değerlendirilmesine olanak sağlayan küresel bir veri tabanı olarak düşünülebilir. Birçok farklı veri formatının kullanılabilmesine olanak sağlayan RDF veri modeli ile farklı Web servislerinin, API'lerin farklı veri yapılarının ortaya çıkardığı veriler arasında bağlantı kuramama, verilerin birlikte çalışamaması gibi sorunlara çözüm sağlanmaktadır. Semantik Web bu özelliğiyle yazılımcıların uygulama geliştirmek için her verinin yapısını bilmek zorunda olması gibi bir sorunu ortadan kaldırmaktadır. Semantik Web'de uygulama geliştirmek için sadece RDF veri modelini bilmek yeterli olmaktadır. Semantik Web veri modeli olan RDF, ayrıca verilerin formatını gözetmeksizin veriler arasında bağlantılar kurulmasını sağlamaktadır. Bu özelliği ile Semantik Web farklı veri setleri arasında ilişkiler kurulmasına yardımcı olur.

Semantik Web, Açık Dünya Varsayımı (Open World Assumption – OWA) ilkesi üzerine kurulmuştur. OWA ilkesi uyarınca; bir ifadenin doğruluğu, o ifadenin herhangi biri tarafından doğru veya yanlış olarak bilinmesinden bağımsızdır. Başka bir şekilde anlatılırsa, bir ifadenin belirtilmemiş olması, o ifadenin olmadığı anlamına gelmez. İlişkisel veri tabanları ise Kapalı Dünya Varsayımına (Close World Assumption – CWA) dayandırılmıştır. CWA verinin sınırlarını, limitlerini belirlerken; OWA tanımı yapılmamış bir kavramın yok sayılamayacağı üzerine yapılır [17].

Şekil 3.1 Geleneksel Web ve Semantik Web'de yaygın olarak kullanılan teknolojiler ve bunların kesişimini; Şekil 3.2 ise Semantik Web mimarisini göstermektedir. Semantik Web'de; gerçek dünya nesneleri ve soyut kavramlar URI'lerle ifade edilir (bkz. Bölüm 3.2), veri setleri OWL, RDFS gibi ontolojilerin sunduğu tanımlamalar (bkz. Bölüm 3.4) ile RDF formatında temsil edilir (bkz. Bölüm 3.3), ve SPARQL teknolojisi (bkz. Bölüm 3.5) ile sorgulanır.

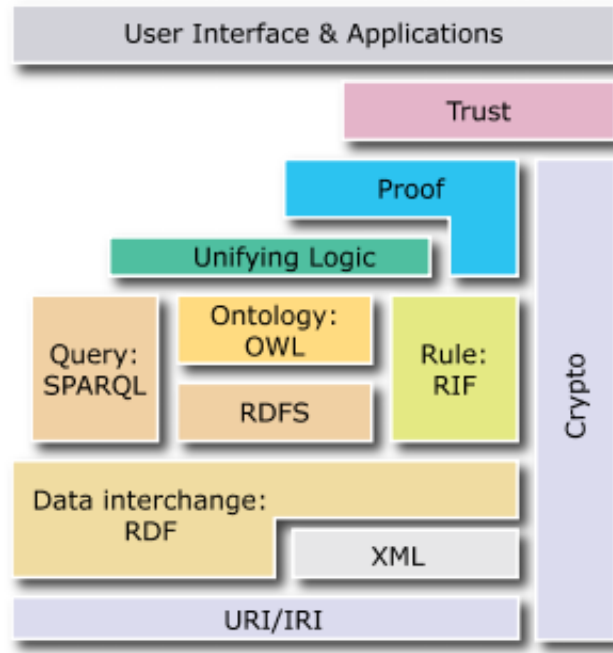


Şekil 3.1 Geleneksel Web ve Semantik Web teknolojileri [55]

3.1 Bağlı Veri İlkeleri

Bağlı Veri terimi, yapılandırılmış verilerin Web’de URI’ler ve RDF kullanılarak sunulması, paylaşılması ve birbirine bağlanmasına ilişkin en iyi uygulamaları tanımlar [14]. Bu yaklaşım, yapılandırılmış verilerin Web üzerinde yayımlanması ve farklı kaynaklardan gelen verilerin etkin biçimde birbirine bağlanmasını sağlar [15]. Bağlı Veri yaklaşımının dört temel ilkesi vardır:

- Şeyleri adlandırmak için URI’leri kullan,
- İnsanların bu adlarla ilgili dokümanlara Web’de erişebilmesi için HTTP URI’leri kullan,
- RDF, SPARQL gibi standartları kullanarak şeylerle ilgili yararlı bilgiler sağla,
- Şeylerle ilgili daha çok bilginin Web’den bulunmasını sağlamak için, RDF belgelerinde, ilgili diğer URI’lere bağlantı kurulmasını sağla [54].



Şekil 3.2 Semantik Web yapısı [56]

Bağlı Veri yaklaşımının birinci ilkesi; klasik Web belgelerinin yanı sıra tüm gerçek dünya nesneleri ve soyut kavramların da URI referansları kullanılarak tanımlanmasını öngörür. Bu bağlamda; örneğin insanlar, coğrafi yerler, araçlar gibi somut şeyler, 'birini tanımak' gibi soyut bir ilişki URI'lerle tanımlanabilir. Geleneksel Web'de sadece dokümanları tanımlamak için kullanılan URI tanımlayıcıları, Semantik Web'de, adı olan her şeyin tanımlanmasında kullanılır.

Web'in küresel erişim mekanizması HTTP protokolleridir. İkinci ilke, URI'lerle tanımlanan gerçek dünya nesnelere, kavramlara ve ilişkilere erişebilmek için HTTP protokolünün kullanımını öngörür. Bu ilke, bir HTTP istemcisinin HTTP protokolü ile bir URI'ye bakıp, bu URI ile tanımlanmış kaynağa ait dokümanlara erişebilmesini sağlar. Kaynaklara ilişkin dokümanlar farklı formatlarda sunulabilir.

Bağlı Veri yaklaşımının üçüncü ilkesi yapılandırılmış dokümanların Web'de sunumu için standart bir veri modelinin kullanımını gerektirir. Bunun için makineler tarafından anlaşılabilen Kaynak Tanımlama Çatkısı (Resource Description Framework – RDF) dilinin kullanımı önerilir. RDF veri modelinde veriler <S, P, O> üçlüsüyle temsil edilen RDF ifadeleriyle temsil edilir. Bu ifadede S özneyi, P ilişkiyi ve O nesneyi ifade eder.

Dördüncü ilke, RDF dokümanlarını sorgulamak için SPARQL dilinin kullanılmasını, veri setlerinin insanlar tarafından da anlaşılabilmesi için HTML olarak da sunulması gerekliliğini vurgular. Geleneksel Web’de sadece Web dokümanlarını ilişkilendiren hiper linkler, Semantik Web ve Bağlı Veri yaklaşımında ilgili olduğu düşünülen her şeyin ilişkilendirilmesinde kullanılır. Bağlı Veri kapsamında bu linkler RDF linki olarak adlandırılır. Bu ilke ile farklı ad alanlarındaki (namespace) URI’ler birbirine bağlanarak, farklı veri setlerindeki kaynaklar birbiriyle ilişkilendirilir. RDF linkleri de yine RDF üçlüsü ile ifade edilir. Bu üçlünün öznesi, veri setinin ad alanındaki bir URI’yi ifade ederken, ilişki ve nesne URI’leri, diğer veri setlerinin ad alanlarındaki URI’leri işaret eder [14].

Şekil 3.3, Bağlı Veri uzayında birbirine bağlanmış olan veri setlerini göstermektedir. Şekildeki her daire, Bağlı Veri olarak yayınlanmış veri setlerini göstermektedir. Daireler arasındaki çizgiler, veri setleri içerisindeki varlıklar arasındaki bağlantıları gösterir. Çizgi kalınlığı bağlantının gücünü ifade eder [14]. Buradaki her daire bilgi tabanı (knowledge base) olarak adlandırılmaktadır. İlk bilgi tabanı, Vikipedi’deki yapılandırılmış verileri Web’de erişilebilir yapmak için oluşturulan DBpedia’dır. DBpedia, Semantik Web için bir merkezsizleşme noktası ve veri setleri arasında ilişkileri kurma amacıyla tasarlanmıştır ve en çok kullanılan veri tabanıdır [57], [58]. DBpedia dışında; Geonames, Freebase⁵, Üst Düzey Eşleşme ve Değişim Katmanı⁶ (Upper Mapping and Binding Exchange Layer - UMBEL) ve OpenCyc⁷ en çok kullanılan bilgi tabanlarındandır. Tüm Bağlı Veri uzayında, birbiriyle bağlantılı olan veri setlerinin yaklaşık % 20’sini konumsal veriler oluşmaktadır. Bir veri seti, bağlantı kurabileceği sınırsız sayıda veri setiyle, RDF linkleri aracılığıyla ilişkilendirilebilir. Örneğin, bir coğrafi konum; bir kişinin evi, iş yeri, bir idari birim veya bir kadastral parsel ile ilişkilendirilebilir. Bağlı Veri, Semantik Web’i bağlanmak için fiilen zorunlu bir uygulamadır [54].

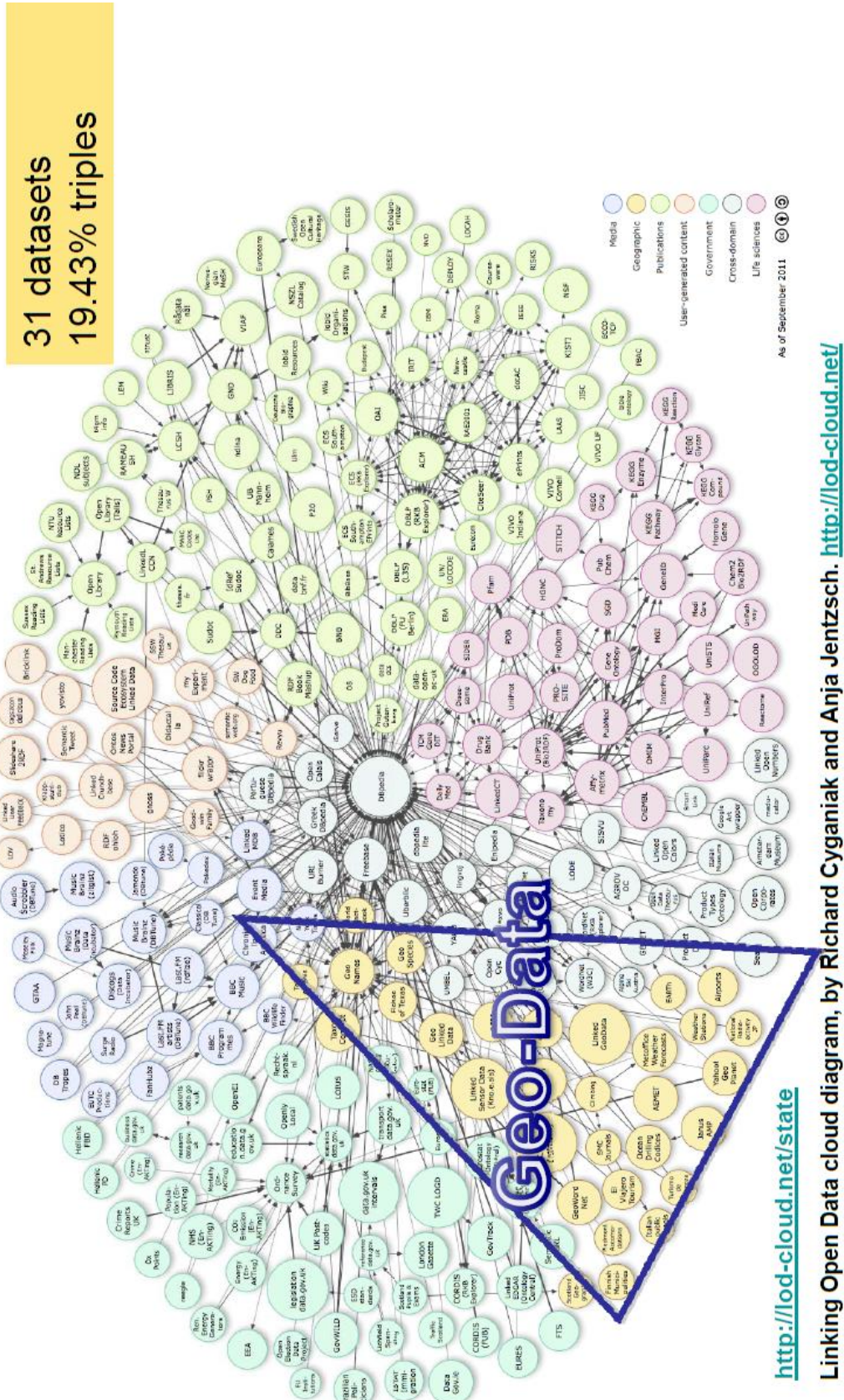
Bağlı Veri; insanlar, şirketler, kitaplar, coğrafi detaylar gibi farklı alanlar arasında bağlantı kurulmasına olanak sağlar. Geleneksel Web’de sabit bir veri seti üzerinde çalışan uygulamaların (mashup) aksine, Bağlı Veri uygulamaları sınırlanmamış bir veri

⁵ <http://www.freebase.com/>

⁶ <http://umbel.org/>

⁷ <http://sw.opencyc.org/>

seti olan küresel veri uzayı üzerine uygulamalarını inşa eder. Bu nedenle daha güvenilir sonuçlar elde edilebilir. Bağlı Veri arama motorları ile Semantik Web’de toplanmış tüm veriler, veri kaynakları arasındaki RDF linkleri izlenerek, günümüzde kullanılan yerel veri tabanı mantığına benzer şekilde anlamlı sorgular yapılabilmesine olanak verecektir.



Şekil 3.3 Semantik Web'de konumsal bilgi içeren Bağlı Veri örnekleri [59]

Bağlı Veri'nin sağlayabileceği yararlar, Birleşik Krallık Konum Programı Raporu [60] ile şöyle listelenmiştir:

- Etkin bir karar verme mekanizması,
- Organizasyonlar arası etkili veri paylaşımı,
- Kolay veri paylaşımı,
- Veri setlerinin yeniden kullanılmasının üst seviyeye çıkarılması,
- Tekrarlı olarak veri toplamının azalması,
- Maliyeti azaltma,
- Veri madenciliği çalışmalarını hızlandırma,
- Büyük verilerle başa çıkabilme kapasitesi,
- Veri tabanlarının izole ve lokal yapısını kaldırıp, veri setlerini birbirine bağlama,
- Farklı uygulamalar için aynı verinin birkaç farklı öznitelik için tekrarlı olarak toplanmasını engelleme,
- Farklı veri yapılarının oluşturduğu farklı çözünürlükte verilerin, koordinat sistemlerinin, sınıflandırma şemalarının, birleşmeyi engelleyen yapısını çözme.

İzleyen bölümlerde Semantik Web ve Bağlı Veri uygulamalarına temel oluşturan; URI, RDF ve SPARQL teknolojileri, genel içerikleriyle açıklanmıştır.

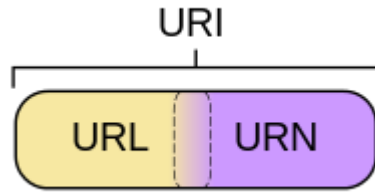
3.2 URI Tanımlama İlkeleri

Geleneksel Web'de herhangi bir konuma erişebilmek için Web Tarayıcısına bir adres yazılır. Bu adres, Birörnek Kaynak Konumlayıcı (Uniform Resource Locator – URL) olarak tanımlanmıştır. Web kullanıcılarının genellikle kullandıkları URL ve URL'nin diğer kaynak tanımlayıcılarla ilişkisi aşağıda açıklanmıştır.

3.2.1 Kaynak Tanımlayıcıları

Birörnek Kaynak Tanımlayıcı (Uniform Resource Identifier – URI); bir kaynağın Web’deki yerini belirten konumlayıcı (locator), bir ad veya her ikisi birden olabilir. URL ise bir kaynağın Web’deki konumunu gösteren bir konumlayıcıdır. URL, URI’nin bir altkümesidir (bkz. Şekil 3.4). URL, URI’nin bir kaynağı tanımlama özelliğinin yanına, kaynağın bir sunumunu veya bir ağdaki konumunu da ekler. Örneğin, <http://www.araziyonetimi.org/> arazi yönetimi kaynağını tanımlayan bir URI’dir, kaynağın bir HTML sunumu olması nedeniyle, aynı zamanda bir URL’dir. Özetle, her URL aynı zamanda bir URI olabilirken, bir URI yalnızca kaynakları adlandıran bir isim olarak kullanılabilir [61], [62], [63].

Birörnek Kaynak İsmi (Uniform Resource Name – URN) ise URI’nin isim olan kısmıdır. URN de, URL gibi URI’nin bir altkümesidir (bkz. Şekil 3.4). URN herhangi bir ağda bir konum belirtmez, bir konumlayıcı olarak kullanılmaz ve bir sunumu da (HTML, RDF vd.) bulunmayabilir. Eşsiz, uluslararası bir isim olarak kullanılır. Günümüzde kullanılmakta olan birçok farklı URN şeması bulunmaktadır. Uluslararası Standart Kitap Numarası (International Standard Book Number – ISBN) ve Uluslararası Standart Seri Numarası (International Standard Serial Number – ISSN) en çok kullanılan URN şemalarındandır. Örneğin, Sigma Dergisinin ISSN numarası 1304-7191 olarak belirlenmiştir. Gösterimi urn:ISSN:1304-7191 olarak yapılır. Benzer şekilde, urn:isbn:0451450523 ise 1968 yılında basılan “The Last Unicorn” isimli kitaba ait ISBN URN’sidir [64].



Şekil 3.4 URI, URL ve URN ilişkisi [64]

URI ve alt sınıfları, Bilgi Değişimi İçin Amerikan Standart Kodlama Sistemi (American Standard Code for Information Interchange – ASCII) olarak adlandırılan karakter tipine uygun olarak üretilmelidir [61]. Bu, uluslararası düzeyde kısıtlamalara ve sorunlara

neden olur. Kaynak tanımlayıcılarında İngiliz alfabesi dışında karakterlerin kullanılamaması koşulu, bazı aynı kökten gelen dilleri bile belirli düzeyde kısıtlamaktadır. Bu sorunu aşabilmek için *Uluslararası Kaynak Tanımlayıcı* (Internationalized Resource Identifier – IRI) şeması geliştirilmiştir. IRI, uluslararası olan ‘Unicode/ISO 10646’ karakter setini kullanmaktadır ve sözdizim formatı URI formatına benzemektedir [65].

Şekil 3.5’te, URL, URI ve IRI arasındaki ilişkileri göstermektedir. URL herhangi bir konumu ifade ederken, URI herhangi bir konumu veya bir isimi ifade edebilir. IRI ise herhangi bir alfabe kısıtlaması olmaksızın herhangi bir konumu veya adı ifade edebilir. URN ise URI veya IRI’nin isim kısmını ifade etmek için kullanılabilir.



Şekil 3.5 URL, URI ve IRI ilişkisi [66]

Bir URI referansı, XML dokümanlarında tanımlanan özellik ve öznitelikleri tanımlamak için kullanılabilir. XML tabanlı dokümanlarda aynı özellikler, öznitelikler tekrarlı olarak yer aldığından ve dokümanlarda büyük yer kapladığından XML Ad Alanı (XML namespace) kavramı Dünya Web Konsorsiyumu (World Wide Web Consortium - W3C) tarafından ortaya atılmıştır. XML Yer Alanı olarak URI ve dolayısıyla IRI, URL ve URN kullanılabilir [67]. Örneğin, RDF tanımlamaların yer aldığı <http://www.w3.org/1999/02/22-rdf-syntax-ns#> URI'sinin ad alanı 'rdf' olarak belirlenmiştir. Bir dokümanda RDF tanımlamaları kullanılacaksa URI yerine sadece ad alanı ve ad alanındaki özellik yazılarak kullanılabilir. XML ad alanındaki her tanımlama evrensel olarak yapılır. Bir ad alanında tanımlanan bir idari birim, o ad alanında sadece bir kez tanımlanabilirken, başka bir ad alanında aynı idari birim tanımlanabilir [68].

Çizelge 3.1 HTML ve RDF dokümanlarında yer alanı örneği

```
<?xml version="1.1"?>
<html:html xmlns:html='http://www.w3.org/1999/xhtml'>

<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF
  xmlns:cc="http://creativecommons.org/ns#"
```

Bir HTML dokümanından örnek vermek gerekirse; Çizelge 3.1'in ilk iki satırında görüldüğü üzere 'http://www.w3.org/1999/xhtml' URI'sine, XML ad alanı olarak 'html' atanmıştır. Bu dokümanın devamında, sözü edilen URI'nin yerine artık sadece "html" XML ad alanı kullanılacaktır. Benzer şekilde, Çizelge 3.'ün geri kalanında, bir RDF dokümanı için 'http://creativecommons.org/ns#' URI'sine, 'cc' XML Yer Alanının atandığı görülebilir. URI'lere kalıcı, değişmeyen XML Ad Alanlarının atanması, ilgili URI'nin temsil ettiği sözlüğün veya ontolojinin bilinirliğini arttırabilir, yaygın kullanılan duruma getirebilir. 'xml', 'rdf', 'rdfs', 'dcterm', 'xsd', 'gml' vd. gibi XML Ad Alanları, kabul görmüş ve yaygın biçimde aynı URI'leri ifade etmek için kullanılmaktadır [68].

3.2.2 Semantik Web'de URI

Önceki bölümde URI'lerin Web'de bir konum ifade edebileceği ve/veya bir ad olarak kullanılabileceği belirtilmişti. Eğer bir URI, Web'de bir konum belirtmiyorsa, başka bir deyişle bir ad olarak kullanılıyorsa URN olarak adlandırılmıştı. Ancak, Semantik Web'de HTTP URI'ler, doğrudan bir isim yerine de kullanılmaya başlanmıştır.

Semantik Web'de bir URI dört farklı şeyi ifade edebilir: Bir ismi, bir soyut kavramı, Web'de bir konumu ve/veya bir doküman tipini [69]. Semantik Web'de insan, ağaç, yol gibi gerçek dünya varlıkları; düşünceleri, duyguları ifade eden soyut kavramlar URI'ler ile ifade edilebilir. URI'ler sözü edilen *şeyleri* tanımlamanın yanı sıra, Web'de bir konumu ve sunulan bilginin doküman tipini (XML, RDF, vd.) de belirtebilir [13], [14]. Yukarıda, URI'lerin isim olan kısmının URN olduğu belirtilmişti; ancak amaç Semantik Web kapsamında Bağlı Veri oluşturmaksa HTTP URI'lerin kullanılması ve URN gibi diğer URI şemalarından kaçınılmasını gerekmektedir [14].

Günümüzde Web kullanıcıları ve sunucuları, Web dokümanlarını görüntülemek için HTTP protokollerini kullanırlar. Bir HTTP istemcisinin, HTTP protokollerini kullanarak

URI'lere erişebilmesi ve URI tarafından tanımlanmış bir kaynağı cevap olarak alabilmesi, yeniden referanslandırılabilme (dereferenceable) olarak adlandırılmaktadır [14], [15]. Semantik Web'de HTTP şemasının kullanılmasının en önemli nedeni, HTTP protokollerinin URI'leri yeniden referanslandırılabilme özelliğidir.

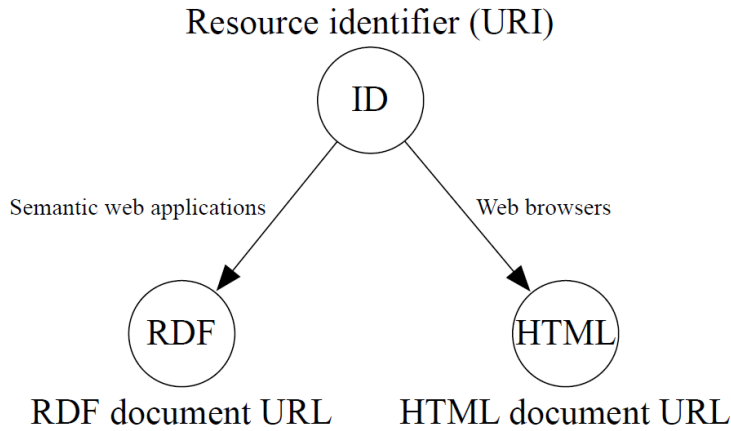
HTTP bunun yanında, aynı Web dokümanının farklı dillerde ve farklı formatlardaki versiyonlarını sunmak için güçlü bir mekanizma sağlar [16], [70]. Bu mekanizma, içerik uzlaştırma (content negotiation) olarak adlandırılır. İçerik uzlaştırma mekanizması, Web istemcisinin 'HTTP headers' özelliğine yolladığı isteğin, sunucu tarafından kontrol edilip uygun bir formatla yanıtlandırarak biçimde tasarlanmıştır [14], [16]. Örnek vermek gerekirse, istemci <http://www.araziyonetimi.org/id/idaribirim/istanbul-besiktas> URI'sini Web tarayıcısı aracılığıyla sunucudan istediğinde, sunucu bu isteğe <http://www.araziyonetimi.org/doc/idaribirim/istanbul-besiktas> olarak cevap verebilir. Cevapta sunucu tarafından belirlenen içerik tipi (content type) *text/html* olarak görülebilir. Sunucudan RDF/XML formatında bilgi isteneceği zamansa, içerik tipi *application/rdf+xml* olarak ayarlanmalıdır [16], [70].

Semantik Web ile birlikte, anlamsal bir genişlemeye uğrayan URI'ler için W3C, 'İyi URI (Cool URI) nasıl olmalıdır?' sorusuna aşağıdaki gibi cevap vermektedir:

- İnsanlar ve makineler, URI'nin tanımladığı kaynak hakkındaki bilgilere, Web yoluyla erişebilmelidirler.
- URI'nin bir Web dokümanını veya kaynağı ifade ettiği kesin olarak bilinmelidir. Bir URI hem gerçek dünya nesnesi, hem de Web dokümanı olarak kullanılamaz [16], [70].

Şekil 3.6, iyi URI olarak ifade edilebilecek bir HTTP URI yapısını göstermektedir. Bir gerçek dünya nesnesi, bir URI ile (ID) tanımlanmış; bir RDF ve bir HTML dokümanının başka URI'lerle bu gerçek dünya nesnesi hakkında veri sunduğu belirtilmiştir. Ancak burada bir sorun ortaya çıkmaktadır: Bir URI aynı anda hem bir gerçek dünya nesnesini hem de bir Web konumunu ve dokümanını belirtmiyorsa, gerçek dünya nesnesini, yani bir kaynağı ifade eden URI'ye, HTTP protokollerini kullanarak nasıl erişebilir? Bu sorunun yanıtı, yukarıda sözü edilen HTTP URI şemasının içerik uzlaştırma özelliğidir. Gerçek dünya nesnelerini yeniden referanslandırılabilir olarak tanımlayabilmek için

izleyen bölümlerde kısaca açıklanan iki farklı yaklaşım uygulanır: 303 URI'leri ve hash URI'leri [14], [16], [17], [70].



Şekil 3.6 Semantik Web’de bir kaynağın URI’si ve tanımlayıcı dokümanı [16]

3.2.3 303 ve Hash URI Yaklaşımı

303 URI yaklaşımında, doğrudan nesnenin kendisini ağ üzerinden göndermek yerine, istemciye bir HTTP yanıt kodu (response code) olan ‘303 see other’ kodu ile gerçek dünya nesnesini tanımlayan Web dokümanının URI’si gönderilir [14], [16]. Gerçek dünya nesnelerini ve soyut kavramları tanımlayan HTTP URI’lerini yeniden referanslandırılabilir yapmak için istemci ve sunucu arasında geçen süreci, Heath ve Bizer (2011) şöyle açıklamaktadır:

İstemci, bir gerçek dünya nesnesini veya soyut bir kavramı tanımlayan HTTP URI’yi, bir tarayıcı aracılığıyla ve ‘http get’ isteğiyle birlikte sunucuya yollar. Eğer istemci Semantik Web uygulaması olan RDF dokümanına erişmek istiyorsa sunucudan RDF/XML sunumunu ister. ‘Accept: application/rdf+xml’ üstbilgisini (header) de istekle birlikte sunucuya gönderir. Tarayıcı ‘Accept: text/html’ cevabını yollar. Böylelikle, sunucu erişilmek istenen URI’nin bir gerçek dünya nesnesi veya soyut bir kavramı olup olmadığını anlar. Sunucu, kaynağın bir sunumunu göndermediği zaman, HTTP ‘303 see other’ yanıt koduyla cevap verir ve istemciye istenen formatta, bir gerçek dünya nesnesini veya soyut bir kavramı tanımlayan Web dokümanının URI’sini gönderir. İstemci, sunucudan gelen bu URI’ye ‘http get’ isteğini tekrar uygular. Sunucu bu isteği,

HTTP 200 ok statü kodu ile cevaplar ve istemciye istenilen dokümanı, istenilen formatta gönderir [14].

Çizelge 3.2 HTTP protokolleri ile 303 URI'sine erişim [14]

```
1 GET /people/dave-smith HTTP/1.1
2 Host: biglynx.co.uk
3 Accept: text/html;q=0.5, application/rdf+xml

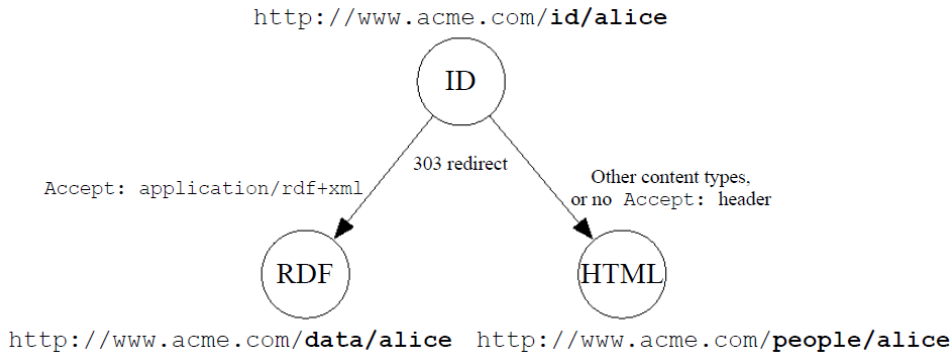
1 HTTP/1.1 303 See Other
2 Location: http://biglynx.co.uk/people/dave-smith.rdf
3 Vary: Accept

1 GET /people/dave-smith.rdf HTTP/1.1
2 Host: biglynx.co.uk
3 Accept: text/html;q=0.5, application/rdf+xml

1 HTTP/1.1 200 OK
2 Content-Type: application/rdf+xml
3
4
5 <?xml version="1.0" encoding="UTF-8"?>
6 <rdf:RDF
7 xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
8 xmlns:foaf="http://xmlns.com/foaf/0.1/">
9
10 <rdf:Description rdf:about="http://biglynx.co.uk/people/dave-smith">
11 <rdf:type rdf:resource="http://xmlns.com/foaf/0.1/Person"/>
12 <foaf:name>Dave Smith</foaf:name>
```

Çizelge 3.2’de, HTTP protokolleri kullanılarak bir 303 URI’sine nasıl erişildiği, uygulamalı olarak anlatılmıştır. İlk adımda istemci, “biglynx.co.uk” alanına (domain) “/people/dave-smith” kişinin RDF/XML formatındaki dokümanını istediğini belirtmiştir. Sunucu bu isteğe ‘303 see other’ yanıt kodu ile yanıt vermiş ve RDF/XML dokümanının bulunduğu konumu belirtip, ikinci kez HTTP get isteğinin uygulanmasını sağlamıştır. Sunucu bu ikinci isteği ‘200 ok’ statü kodu yanıtlayarak, istemcinin istediği RDF/XML formatındaki dokümanı sunmuştur.

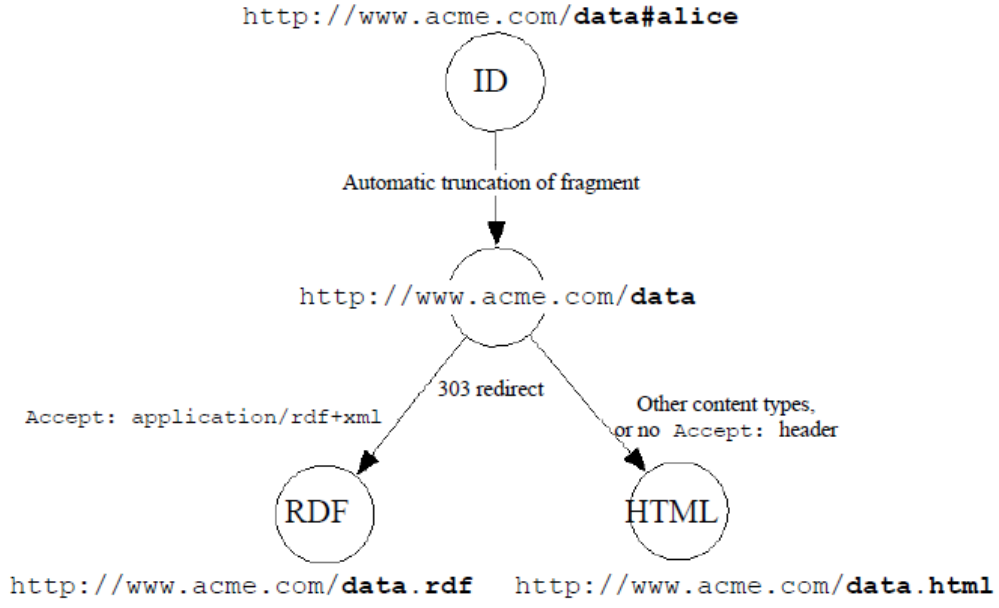
Şekil 3.7’de örnek bir 303 URI modeli verilmiştir. <http://www.acme.com/id/alice> URI’si ‘alice’ adında bir kişiyi (gerçek dünya nesnesi) tanımlamaktadır. Bu URI, bir tarayıcı yardımıyla istendiğinde, tarayıcı ‘303 see other’ kodu ile yanıt vermektedir. Eğer bir RDF/XML dokümanı isteniyorsa sunucu ilk istenen URI’yi <http://www.acme.com./data/alice> URI’sine yönlendirecek; eğer HTML sunum isteniyorsa sunucu <http://www.acme.com/people/alice> URI’sine yönlendirecek şekilde ayarlanır [16], [70].



Şekil 3.7 303 URI Çözümü [16]

Gerçek dünya nesnelerinin yeniden referanslandırılabilir olarak tanımlanabilmesi için uygulanan diğer yaklaşım da Hash URI'lerinin kullanımıdır. URI'nin bir kısmının geri kalanından hash sembolü (#) ile ayrıldığı URI'lere Hash URI denir [16]. URI'nin ayrılan kısmı, bölüm kimliği (fragment identifier) olarak adlandırılır [71]. İstemci bir hash URI'ye erişmek istediği zaman, HTTP protokol, URI'nin bölüm kimliği kısmını ana kısımdan ayırıp sunucuya istek yapar. Bunun anlamı, Hash içeren bir URI'nin tamamına doğrudan yanıt alınamayacağıdır. Bu nedenle, Hash URI'ler Web dokümanlarını tanımlamak için kullanılamazken; gerçek dünya nesnelerini ve soyut kavramları tanımlamada karışıklık yaratmadan kullanılabilirler [14], [16], [17]. Hash URI'ler, RDF/XML dokümanlarına yönlendirmede '302 Found', '303 See Other' ve '307 Temporary Redirect' HTTP protokollerini kullanırlar [14], [70].

Şekil 3.8'de içerik uzlaştırma ile birlikte Hash URI'lerinin kullanımı görülmektedir. İlk kısımda <http://www.acme.com/data#alice> URI'si sunucudan istendiğinde; sunucu, URI'nin bölüm kimliği kısmını ayırıp, sadece <http://www.acme.com/data> bölümü için yanıt göndermektedir. Daha sonra, içerik uzlaştırma ile <http://www.acme.com/data> ilgili RDF ve HTML sunumlarına yönlendirilmektedir. Hash URI'lerin Semantik Web kapsamındaki bazı uygulamalarda '303 See Other' yönlendirmesi olmadan da kullanımı olanaklıdır.



Şekil 3.8 İçerik uzlaştırma ve hash URI çözümü [70]

303 ve Hash URI yaklaşımlarının görece üstünlükleri bulunmaktadır. Hash URI'lerinin kullanımı, sunucuya yapılan HTTP isteklerinin sayısını azaltır; böylelikle veriye hızlı erişim sağlar [14]. Hash'tan önceki kısım bir veri setindeki tüm veriler için sabit olduğundan, tüm veri seti tek bir istekle elde edilebilir. Bu nedenle, istemci tüm veriyi aynı anda görüp, yeni bilgiler elde edebilir. Hash URI'lerde sunucu, içerik uzlaşmasına uygun ayarlanmak zorunda değildir [16]. 303 URI'leri ise Hash URI'lere göre daha esnek bir yapıya sahiptir. 303 URI'lere yaygın olarak yapılan eleştiri, gerçek dünya nesnesiyle ilgili Web dokümanlarını tek bir seferde cevap olarak geri alabilmek için sunucuya iki kez HTTP get isteği gönderme zorunluluğu, yani içerik uzlaşmasının uygulanma zorunluluğudur. 303 URI'lerde çok sayıda yönlendirme yapılması, veriye erişimde gecikmelere neden olabilir [14].

Bir uygulamada hangi URI yaklaşımının kullanılması gerektiği, bazı değişkenlere bağlıdır. Hash URI'lerinin, daha çok küçük projeler ve küçük veri setleri için kullanılması daha uygun görülmektedir. 303 URI'leri ise büyük veya büyüyecek veri setleri için uygundur. En geniş kapsamlı Bağlı Veri uygulaması olan DBpedia, 303 URI yaklaşımını seçmiştir. Bunun yanında RDF belgeleri için sözlük terimleri oluşturan çalışmalarda, binlerce terimi ayrı ayrı çağırmak yerine tek bir seferde çağırmak için Hash URI'ler

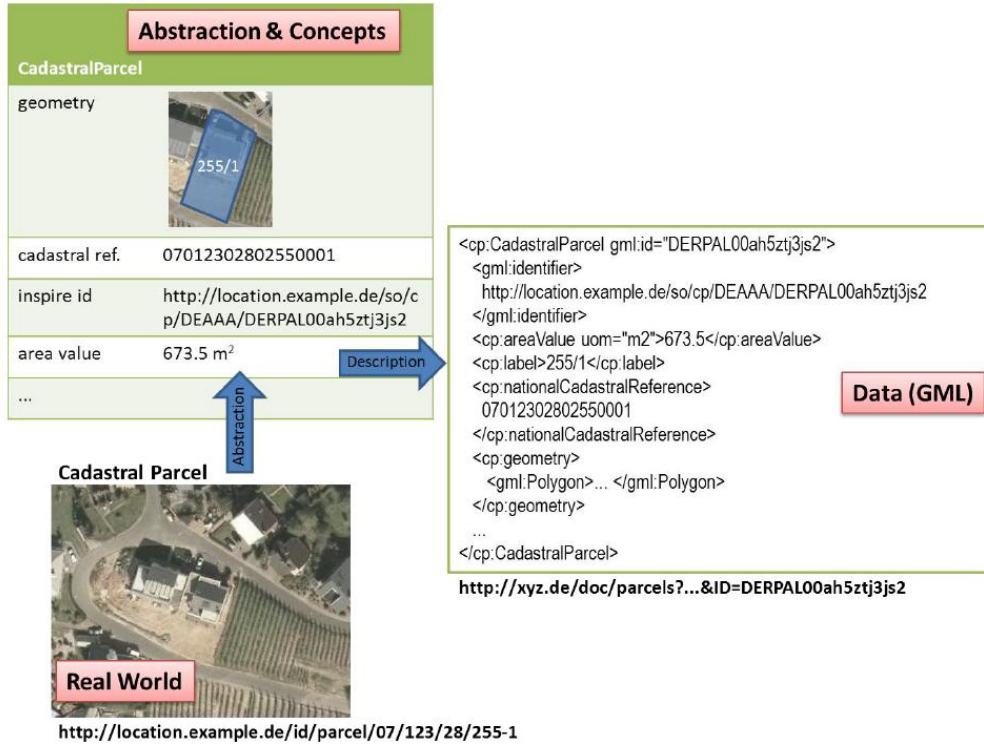
kullanılmaktadır. Örneğin, Dünya Jeodezik Sistemi (World Geodetic System - WGS84), SKOS gibi sözlükler Hash URI'leri kullanmaktadır [14]. Eğer bir çalışmada hangi URI yaklaşımının kullanılacağına karar verilemiyorsa, daha esnek koşullar sunan 303 URI'lerinin kullanılması önerilmektedir [16], [17].

3.2.4 Küresel URI Modelleme Çalışmaları

3.2.4.1 INSPIRE URI Modeli

INSPIRE veri temalarına temel oluşturan jenerik kavramsal modelde, konumsal nesneler ve kod listeleri gibi yapıların URI'ler ile tanımlanması önerilmiş; URI modelleme kuralları belirlenmiştir. INSPIRE'da konumsal nesne tanımlayıcısı olarak URI'lerin kullanımı, konumsal nesneleri belirlemek, izlemek, yeniden kullanıma olanak tanımak ve başka veri setleriyle bağlantılar kurulmasına olanak sağlar. INSPIRE kapsamında, konumsal nesne (spatial object) ve konumsal şeyler (spatial thing) için ayrı modelleme kuralları belirlenmiştir. INSPIRE bağlamında konumsal nesne, özel bir konumla ilişkili bir gerçek dünya nesnesinin soyut sunumudur. Konumla ilişkili olan herhangi bir şey ise konumsal şey olarak tanımlanmıştır. Özetle, konumsal nesneler, gerçek dünya nesnelerinin modelidir.

INSPIRE, ulusal düzeyde belirlenecek URI'ler için şu şablonun kullanımını önermektedir: <http://{subdomain}.{domain}/{type}/{metadata}/{authority}/{localid}/{version}>. Bu modelin “type” kısmında kullanılabilecek kısaltmalardan; ‘id’ gerçek dünya nesnelerini, ‘so’ konumsal nesneleri, ‘doc’ konumsal nesne ile ilgili dokümanları ve ‘def’ ontolojileri tanımlar. Şekil 3.9’da INSPIRE tarafından kadastral parsel için yapılan örnek URI tanımlamaları görülmektedir. Gerçek dünyadaki kadastral parsel (spatial thing) <http://location.exmaple.de/id/parcel/07/123/28/255-1> URI’si; parselin modelini ifade eden (spatial object) URI ise <http://location.example.de/so/cp/DEAAA/DERPAL00ah5ztj3js2> olarak belirlenmiştir. Parselle ilgili bilgiler sunan GML dokümanı, <http://xyx.de/doc/parcels?...&ID=DERPAL00ah5ztj3js2> olarak ifade edilmiştir [72].



Şekil 3.9 INSPIRE kadastral parsel URI tanımlaması [72]

3.2.4.2 Kamu Sektörü İçin URI Setleri Tanımlama

Birleşik Krallık’da 2010 yılında yayımlanan ‘Yönetimler Arası Kurumsal Bilişim Mimarisi’ (cross-Government Enterprise Architecture) başlıklı raporda, Birleşik Krallık Harita Kurumu (Ordnance Survey – OS) ve birkaç farklı kurum dışında verilerini Açık Bağlı Veri olarak yayımlayan kurum olmadığı belirtilmiş; Birleşik Krallık Kamu Sektörü için data.gov.uk alanının kullanılması ve eğitim, sağlık vb. gibi veri temaları için URI setlerinin geliştirilmesi hedeflenmiştir. Bu amaçla, ‘Birleşik Krallık Kamu Sektörü için URI Setlerinin Tasarlanması’ başlıklı bir rapor hazırlanmıştır [73].

Raporda, Semantik Web kapsamında beş farklı tip URI’nin kullanılabileceği belirtilmektedir. Bunlar; gerçek dünya nesnelerini tanımlayan URI’ler, gerçek dünya nesneleri hakkında Web dokümanlarını ifade eden doküman URI’leri ve sunum URI’leri, bir veri setindeki tüm URI’leri listeleyen liste URI’leri ve ontolojilerde kullanılan kavramlar ile kavramlar arası ilişkileri tanımlayan URI’ler olarak belirlenmiştir. Rapor kapsamında tüm bu URI tipleri için standart bir model oluşturmuştur. Şekil 3.10’da görülen bu örnekte, URI’lerde gerçek dünya nesneleri için ‘id’, bu nesnelerin Web dokümanları için ‘doc’, kavram tanımlamaları için ‘def’ ve veri setlerini belirtmek için

‘set’ adında dizinler URI’lerde kullanılmıştır. Bu tanımlamalar, INSPIRE kapsamında yapılan URI tanımlamalarıyla örtüşmektedir. Makinelerin anlayabileceği Web dokümanları için ise farklı URI’ler üretilmesi gerekliliği vurgulanarak; URI’lerin sonuna dosya tipinin (örn., RDF, CSV, JSON) belirtilmesi önerilmiştir [73].

Raporda, URI setleri oluşturma işinin sektörlere bölünmesi, her URI’nin ilgili sektör tarafından oluşturulması gerektiği belirtilmektedir. Oluşturulan veri setleri için lisans, doğruluk, köken gibi bilgilerin Bağlantılı Veri Setleri Sözlüğü (Vocabulary of Interlinked Datasets – VOID) gibi tanımlama şemaları kullanılarak belirtilmesi önerilmiştir.

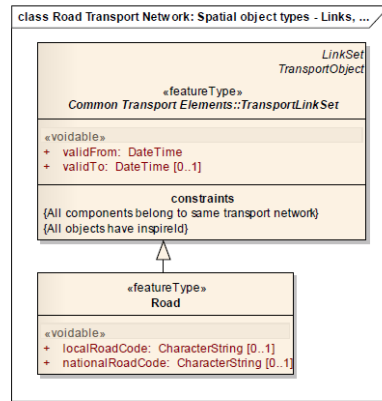
URI Type	URI structure	Examples
Identifier	http://{domain}/id/{concept}/{reference} or http://{domain}/id/{concept}/{reference}#id	http://education.data.gov.uk/id/school/78 http://education.data.gov.uk/school/78#id http://transport.data.gov.uk/id/road/M5/junction/24
Document	http://{domain}/doc/{concept}/{reference}	http://education.data.gov.uk/doc/school/78
Representation	http://{domain}/doc/{concept}/{reference}/{doc.file-extension}	http://education.data.gov.uk/doc/school/78/doc.rdf
Definition of the scheme concept	http://{domain}/def/{concept}	http://education.data.gov.uk/def/school
List of scheme identifiers	http://{domain}/doc/{concept}	http://education.data.gov.uk/doc/school
Set	http://{domain}/set/{concept}	http://education.data.gov.uk/set/school

Şekil 3.10 İngiltere kamu sektörü URI modelleri [73]

Birleşik Krallık Kamu Sektörü için URI Setlerinin Tasarlanması başlıklı raporda ayrıca, INSPIRE kapsamında tanımlanan konumsal nesneler için de URI tanımlama ilkeleri belirlenmiştir. Bu ilkeler, INSPIRE veri temalarında kullanılan tematik tanımlayıcı ve eşsiz nesne tanımlayıcı tanımlamalarını aynen kullanmıştır. Gerçek dünya nesneleri (konumsal şey) için <http://{sector}.data.gov.uk/id/{concept}/{reference}/{version}>; konumsal şey’lere ilişkin dokümanlar için <http://{sector}.data.gov.uk/doc/{concept}/{reference}/{version}>; ontolojik tanımlamalar için <http://{sector}.data.gov.uk/def/{concept}/{term}> ve konumsal nesne tanımlaması için <http://{authority}/so/{theme}/{class}/{namespace}/{localId}/{versionId}> URI şablonlarının kullanılmasını önerilmiştir. Bu URI modellerinde ‘sector’, URI’nin ilgilendirdiği veri temasını veya veri setini ifade etmektedir. URI’de bir ‘id’ ifadesi varsa

gerçek dünya nesnesinin; 'doc' varsa gerçek dünya nesneleri ile ilgili dokümanların; 'def' varsa tanımlamaların ve 'so' varsa gerçek dünya nesnelerinin bir modelini belirtilmektedir. 'concept' sektörü, INSPIRE kapsamında veri temasını ve sınıfını; 'reference' belirlenen sektördeki örneklemeleri; 'version' versiyonları ifade etmektedir. Örneğin, <http://transport.data.gov.uk/id/road/M5> URI'si ile ulaşım (transportation) sektöründeki M5 referansına sahip bir gerçek dünya nesnesi tanımlanmıştır.

Şekil 3.11'de INSPIRE Ulaşım Veri Temasının bir bölümü için hazırlanmış URI tanımlamaları görülmektedir. Örneğin, <http://location.data.gov.uk/def/tn/road/natioanlRoadCode> URI'si INSPIRE Ulaşım veri modelindeki 'Road' sınıfında tanımlanan 'nationalRoadCode' özniteliğini belirten bir tanımlamadır.



URI	Description
http://location.data.gov.uk/def/tn/road/Road	Road featureType
http://location.data.gov.uk/def/tn/road/localRoadCode	localRoadCode property of a Road feature
http://location.data.gov.uk/def/tn/road/nationalRoadCode	nationalRoadCode property of a Road feature
http://location.data.gov.uk/def/inspire/validFrom	INSPIRE wide reusable validFrom property
http://location.data.gov.uk/def/inspire/validTo	INSPIRE wide reusable validTo property
http://location.data.gov.uk/def/inspire/inspireId	INSPIRE wide reusable inspireId property

Şekil 3.11 INSPIRE ulaşım veri teması örnek URI tanımlamaları [73]

3.2.4.3 ISA Kalıcı URI Tanımlama Çalışmaları

Avrupa Komisyonu, Avrupa Kamu Yönetimi İçin Birlikte Çalışabilirlik Çözümleri (Interoperability Solutions for European Public Administration – ISA) kapsamında, Semantik Birlikte Çalışabilirlik Komitesi (Semantic Interoperability Community – SEMIC) tarafından güncel URI seti belirleme çalışmaları incelenmiş ve kalıcı URI'ler belirlemek için öneriler sunulmuştur [74]. Bu çalışma kapsamında, kalıcı URI seti oluşturmaya ilişkin diğer girişimlerin sonuçları da dikkate alınmıştır. Bu çalışmalardan Avrupa

Komisyonu Enformel Çalışma Grubunun hazırladığı kalıcı URI belirleme çalışmalarında varılan sonuçlar, diğer tüm çalışmaları kapsar nitelikte olduğundan ilgi çekicidir. Çalışma Gurubunun vardığı sonuçlara göre URI'ler:

- Her kaynağı tanımlamalı,
- Kalıcı ve sabit olmalı,
- Yönetilebilir olmalı,
- Eşsiz olmalı,
- Açık ve kısa olmalı,
- Birbiriyle bağlantılı olmalı,
- İnsanların anlayabileceği şekilde oluşturulmalı, örneğin, europa.eu/health/1235564798765465498 yerine, europa.eu/health/guidelines kullanılmalı,
- Yapı ve format olarak tutarlı olmalı, örneğin, europa.eu/healthy/ yerine, europa.eu/health/ kullanılmalı,
- Anahtar sözcük içermemeli,
- Dosya uzantısı içermemeli,
- 'www' içermemeli,
- Küçük harflerle oluşturulmalı,
- Boşluk veya vurgulama içermemeli,
- !"£\$%^&*() karakterleri yerine tire veya altçizgi içermelidir.

Çalışma gurubu, genel URI belirleme çalışmaları için standart bir dizinin belirlenmesini ve dizinde oluşturulacak her bir alan için sorumluluk listeleri oluşturulmasını öngörmüştür. Bu yönetim, sorumluluk listelerinin EuroVoc⁸ gibi bir standart taksonomi ile bağlantılı hale getirilmesi gerektiği belirtilmiştir. Raporda, W3C tarafından

⁸ <http://eurovoc.europa.eu/>

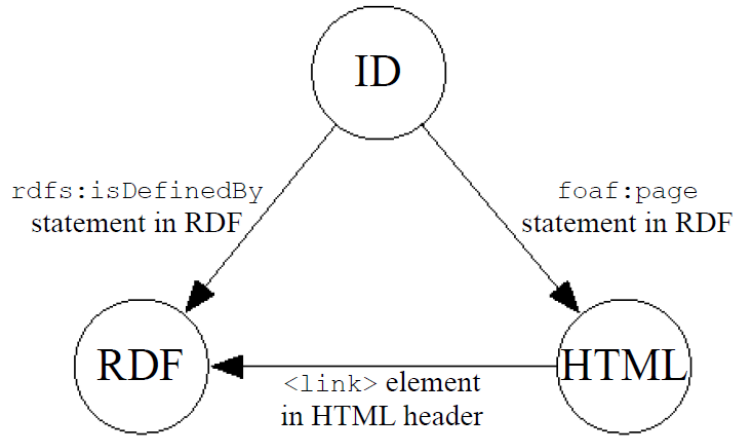
yayımlanan URI belirleme ilkelerine bağılı kalınmasını önermiştir. Rapor, kalıcı URI'lerin tanımlanmasına ilişkin 10 kural belirlenmiştir:

- Raporda, <http://{domain}/{type}/{concept}/{reference}> şablonunun kullanılması önerilmektedir (URI'deki "type" kısmı id, doc, def, set tanımlamalarını ifade etmektedir).
- Var olan tanımlayıcılar yeniden kullanılmalıdır. Örneğin, bir okul için 123456 tanımlayıcısı atanmışsa URI <http://education.data.example/id/school/123456> şeklinde olmalıdır.
- Birden fazla formatta sunum hazırlanmalıdır. Örneğin, <http://data.example.org/doc/foo/bar.html> ve <http://data.example.org/doc/foo/bar.rdf> URI için HTML ve RDF sunumlarını ifade eder. Ayrıca bu iki URI birbiriyle bağlantılı hale getirilmelidir.
- Gerçek dünya nesneleri için 303 yönlendirmesi kullanılmalıdır.
- Özel bir servis alanı kullanılmalıdır. Örneğin, data.gov.uk, publications.europa.eu gibi özel bir alan, URI'ler için ayrılmalıdır.
- Proje veya kurum adı belirtilmemelidir.
- Versiyon numarası kullanılmamalıdır.
- Eşsiz tanımlayıcılar oluşturmak için, sayı içeren tanımlayıcılarda otomatik artırımının kullanımından kaçınılmalıdır.
- "?param=value" gibi sorgu terimleri kullanılmamalıdır.
- Tanımlayıcılar için dosya uzantısı kullanılmamalıdır [74].

3.2.5 URI'lerin Bağlantılı Hale Getirilmesi

Gerçek dünya nesnesi ile ilgili tüm URI'ler, yani gerçek dünya nesnesini ifade eden kaynak tanımlayıcısı, ilgili RDF dokümanının URI'si ve Web dokümanının URI'si birbirleriyle açıkça bağlantılı hale getirilmelidir [70]. W3C tarafından bu ilişki aşağıdaki gibi örneklenmiştir: <http://www.acme.com/id/alice> URI'si gerçek dünya nesnesini, <http://www.acme.com/people/alice> URI'si HTML sayfasını,

<http://www.acme.com/data/alice> URI'si ise aynı nesnenin RDF dokümanını ifade etmektedir. Kaynağın RDF dokümanında HTML sayfası Kişisel İlişkiler Sözlüğünün (Friend of a Friend – FOAF) özelliği olan foaf:page ile gösterilmesini önermiştir. Ayrıca, kaynağın HTML dokümanının başlığında (header) bir <link /> elemanı içermesi gerektiği belirtilmiştir. Bu eleman, ilgili RDF dokümanının yerini HTML içinde gösterir. HTML <head> tagları arasında <link "rel="alternate" type="application/rdf+xml" href="http://www.acme.com/data/alice"/> tanımlamaların yerleştirilmesi, RDF dokümanlarının arama motorları aracılığıyla keşfedilmesini sağlar. Eğer Web sayfasındaki verilerle, RDF dokümanındaki veriler birbirinden farklılaşıyorsa "rel="alternate"" özelliği yerine "rel="meta"" özelliğinin kullanılabilir [16] [70]. Şekil 3.12'de gerçek dünya nesnesi ile HTML ve RDF dokümanları arasında kurulması gereken ilişkiler görülmektedir [16].



Şekil 3.12 HTML ve RDF dokümanları arasında bağlantı [16].

Bu bölümde, Bağlı Veri ilkelerinin ilk ikisiyle önerilen URI ve HTTP URI'lerine ilişkin açıklamalar verilmiştir. İzleyen bölümde, Bağlı Veri ilkelerinin üçüncüsü ve Bağlı Veri yayımlama standardı olan RDF veri modeli açıklanmaktadır.

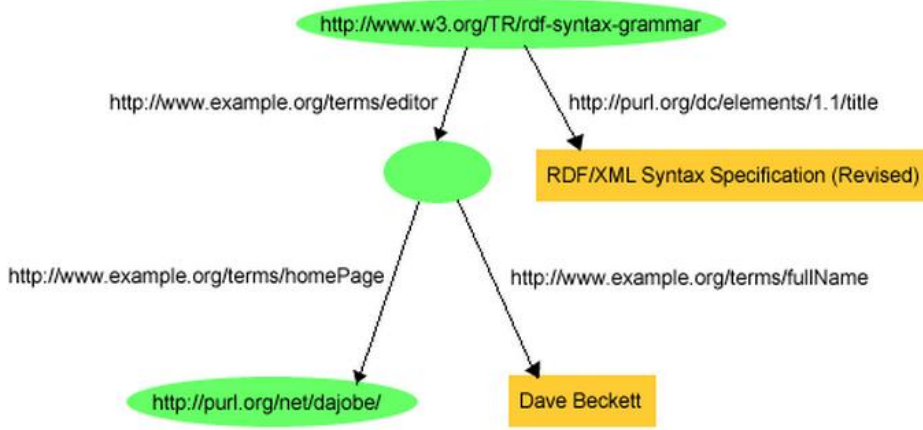
3.3 RDF Veri Modeli

Geleneksel Web'de veri setleri farklı yapılarda oluşturulmuştur. Bu nedenle, herhangi bir kaynağın verilerinin başka bir uygulamada kullanılmasını sağlamak için veri

yapılarının bilinmesi gerekir. Farklı kaynakların sunduğu farklı veri yapısını öğrenmek, Web programcılarının uzun zamanını alır; Web'in kullanım kapasitesini ve uygulamaların sürdürülebilirliğini kısıtlar. Semantik Web yaklaşımı bu soruna çözüm sunar. Web içeriğinde bulunan geniş yelpazedeki birçok uygulama, sürdürülebilirliklerine olanak sağlamak için standart bir içerik modeli olan RDF veri modeli ile temsil edilebilir [14]. RDF veri modelinin amacı, Web kaynaklarının tanımlanması için kavramsal bir çerçeve sunarak, heterojen veri yapılarını düzenlemek ve Web kaynaklarının herhangi bir alandan (domain) bağımsız tanımlanmasını sağlamaktır [14], [17]. RDF veri modeli, HTML'nin aksine makineler tarafından anlaşılabilir yapıdadır [15]. Amaç, kaynak verilerinin farklı uygulamalarda kullanılmasını sağlamak ve birlikte çalışabilirliği kolaylaştırmaktır [17]. 1999 yılında ilk taslağı çıkan RDF veri modeli, 2004 yılında W3C tarafından RDF 1.0 versiyonu olarak standartlaştırılmıştır. 2014 yılında ise yapılan yeni tanımlamalar ve güncellemelerle RDF 1.1 versiyonu sunulmuştur [75], [76].

RDF, XML tabanlı bir veri modelidir, ancak kesinlikle bir XML türü değildir. RDF veri modelinin ortaya çıkmasının nedenleri; XML'in yapısal karışıklığı ve çeşitliği, XML dokümanını anlamak için bir XSD dokümanına duyulan ihtiyaç ve XML dokümanlarını sorgulamanın RDF dokümanlarını sorgulamadan daha zor olması olarak özetlenebilir [77]. RDF veri modelinin ağaç (tree) şeklindeki yapısı bilgisayarlar tarafından XML'den daha kolay anlaşılabilir olduğundan RDF, XML'e göre kolay sorgulanabilir yapıdadır.

RDF veri modeli, düğümler (node) ve çizgilerden (arcs) oluşan RDF çizgeleri (graph) sunar [14], [75]. Bir RDF çizgesi, bir veya birden çok RDF ifadelerinden (statements) oluşur. Bir RDF ifadesi, özne, yüklem ve nesneden (S, P, O) meydana gelir. Yüklem sadece URI'lerle tanımlanabilirken; özne ve nesne ise URI'lerle, literal olarak veya boş düğüm olarak tanımlanabilir. Nesnesinin literal olduğu üçlülere RDF literal adı verilir [14]. Şekil 3.13'de örnek bir RDF grafiği yapısı görülmektedir. RDF ifadesindeki özne ve nesne, RDF çizgesinde düğüm; yüklem ise çizgilerle temsil edilmiştir [76].



Şekil 3.13 Bir RDF çizgesi örneği [76]

Yukarıda da belirtildiği gibi, bir RDF çizgesindeki düğümler ve çizgiler; URI olarak, sayı, tarih, yazı vd. gibi sözel ifadeler veya boş düğüm (blank node) olarak oluşturulabilir. Düğüm ve çizgilerde literal (örn., sayı, tarih, ölçü birimi vd) kullanılacak ise, literal değerın nasıl tanımlanacağını belirten standardize edilmiş veri tipleri, şema ad alanları ile birlikte kullanılmalıdır. Örneğin, literaller, XSD’de tanımlanmış ve Şekil 3.14’te görülen değerleri alabilir. Boş düğümlerin ise herhangi bir tanımlayıcısı ve URI’si yoktur [14], [75], [76]. Boş düğümlere RDF çizgeleri dışında erişilemez ve bu yüzden boş düğümler, farklı kaynaklardan gelen verilerin entegrasyonunu sağlamaz [14], [17].

	Datatype	Value space (informative)
Core types	xsd:string	Character strings (but not all Unicode character strings)
	xsd:boolean	true, false
	xsd:decimal	Arbitrary-precision decimal numbers
	xsd:integer	Arbitrary-size integer numbers
IEEE floating-point numbers	xsd:double	64-bit floating point numbers incl. $\pm\text{Inf}$, ± 0 , NaN
	xsd:float	32-bit floating point numbers incl. $\pm\text{Inf}$, ± 0 , NaN
Time and date	xsd:date	Dates (yyyy-mm-dd) with or without timezone
	xsd:time	Times (hh:mm:ss.sss...) with or without timezone
	xsd:dateTime	Date and time with or without timezone
	xsd:dateTimeStamp	Date and time with required timezone
Recurring and partial dates	xsd:gYear	Gregorian calendar year
	xsd:gMonth	Gregorian calendar month
	xsd:gDay	Gregorian calendar day of the month
	xsd:gYearMonth	Gregorian calendar year and month
	xsd:gMonthDay	Gregorian calendar month and day
	xsd:duration	Duration of time
	xsd:yearMonthDuration	Duration of time (months and years only)
	xsd:dayTimeDuration	Duration of time (days, hours, minutes, seconds only)
Limited-range integer numbers	xsd:byte	-128...+127 (8 bit)
	xsd:short	-32768...+32767 (16 bit)
	xsd:int	-2147483648...+2147483647 (32 bit)
	xsd:long	-9223372036854775808...+9223372036854775807 (64 bit)
	xsd:unsignedByte	0...255 (8 bit)
	xsd:unsignedShort	0...65535 (16 bit)
	xsd:unsignedInt	0...4294967295 (32 bit)
	xsd:unsignedLong	0...18446744073709551615 (64 bit)
	xsd:positiveInteger	Integer numbers >0
	xsd:nonNegativeInteger	Integer numbers ≥ 0
	xsd:negativeInteger	Integer numbers <0
	xsd:nonPositiveInteger	Integer numbers ≤ 0
Encoded binary data	xsd:hexBinary	Hex-encoded binary data
	xsd:base64Binary	Base64-encoded binary data
Miscellaneous XSD types	xsd:anyURI	Absolute or relative URIs and IRIs
	xsd:language	Language tags per [BCP47]
	xsd:normalizedString	Whitespace-normalized strings
	xsd:token	Tokenized strings
	xsd:NMTOKEN	XML NMTOKENs
	xsd:Name	XML Names
	xsd:NCName	XML NCNames

Şekil 3.14 RDF'ye uygun XSD tipleri [75]

RDF oluşturulurken farklı RDF sözlüklerinden faydalanılması gerektiği gibi, RDF'nin kendi ad alanında (rdf) oluşturduğu URI referanslarını da kullanmak önemlidir. RDF sözlüğünde `rdf:nodeID` ve `rdf:parseType="Resource"` özellikleri boş düğümleri tanımlamak için oluşturulmuştur. `rdf:nodeID` ile herhangi bir URI referansı olmayan boş düğümleri lokal sistemde ifade edilebilir yapmak amaçlanmıştır. `rdf:nodeID`, aynı boş düğümün veri seti içinde tekrar kullanılması gerektiğinde aynı tanımlayıcı ile kullanılmasına olanak sağlar. `rdf:parseType="Resource"` özneliği ise boş düğümün bulunduğu veri setinde tanımlamaların yapıldığı `rdf:Description` çiftlerinin tekrar kullanılması önlemek için kullanılmaktadır. `rdf:ID`, kaynak RDF'de ilk kez tanımlanıyorsa, `rdf:about` ve `rdf:resource` ise RDF dokümanında daha önce tanımlamış kaynağı ifade etmek için kullanılır. Bunun yanında `xml:base` tanımlaması da daha önce yapılan

ontoloji tanımlamaların direkt olarak RDF dokümanında kullanılmasına olanak sağlar. `rdf:about`, `rdf:resource` ve `xml:base` özellikleri RDF’de kaynakları tanımlamak için kullanılan URI’lerin kısaltılma olarak RDF’de ifade edilmesine olanak sağladığından ve RDF doküman boyutunu düşürdüğünden, bu özelliklerin kullanılması önerilmektedir [75]. RDF sözlüğünde ayrıca özellikleri ve özellik elemanlarını gruplamak, sınıflandırmak, dizmek (containers) için bir dizi özellik belirlemiştir. Bunlardan `rdf:Bag` düzensiz bir liste oluşturmada, `rdf:Seq` alfabetik liste düzeninde öznitelikleri listelemede kullanılabilir. `rdf:Alt` özelliği ise alternatif değerleri listelemek için kullanılabilir. Listelemede her bir özellik elemanı `rdf:li` ve `rdf:_n` özellikleriyle ifade edilmektedir. RDF’de özellikleri ve özellik elemanları gruplandırmak için bir de `rdf:parseType="Collection"` özniteliği vardır. Bu öznitelik öncekilerden farklı olarak sadece bu listedeki özniteliklerden bir seçim yapılmasını istemektedir. Herhangi bir alternatif değer veya başka bir özniteliğin bulunmadığı; `rdf:parseType="Collection"` özniteliği görülünce anlaşılabilir. RDF’de özellikleri veya özellik elemanlarını sınırlandırmak için; `rdf:List`, `rdf:first`, `rdf:rest` ve `rdf:nil` gibi özelliklerin de bulunduğu unutulmamalıdır. Bunların dışında `rdf:datatype` özelliğinin, kaynağı yazı ile ifade etmede `rdf:langString`; sayısal değerleri ifade etmek içinse `rdf:value` özelliği kullandığını belirtmek gerekir [75], [76], [78], [79].

Çizelge 3.3 RDF/XML örneği [76]

```

1. <?xml version="1.0"?>
2. <rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
3.     xmlns:dc="http://purl.org/dc/elements/1.1/"
4.     xmlns:exterms="http://example.org/stuff/1.0">
5.   <rdf:Description rdf:about="http://www.w3.org/TR/rdf-syntax-grammar">
6.     <dc:title>RDF/XML Syntax Specification (Revised)</dc:title>
7.     <exterms:editor rdf:nodeID="abc"/>
8.   </rdf:Description>
9.   <rdf:Description rdf:nodeID="abc">
10.    <exterms:fullName>Dave Beckett</exterms:fullName>
11.    <exterms:homePage rdf:resource="http://purl.org/net/dajobe"/>
12.   </rdf:Description>
13. </rdf:RDF>

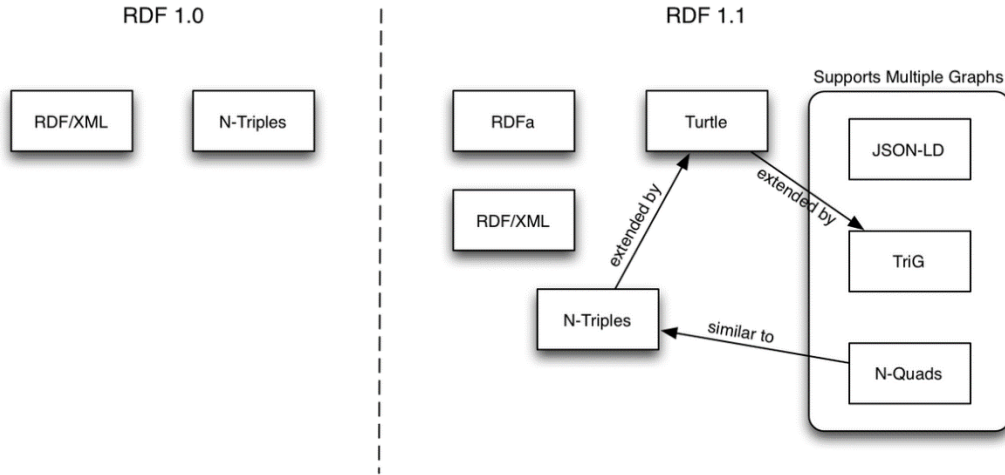
```

Çizelge 3.3’te bir RDF/XML örneği görülmektedir. Daha önce bahsedildiği gibi, RDF dokümanları özelleşmiş birer XML dokümanlarıdır ve bu yüzden XML tanımlamaları ile başlar. Birinci satırda görülen XML tagları bu nedenle her RDF dokümanının başında

bulunmaktadır. Oluşturulacak dokümanın RDF dilinde olduğunu belirtmek içinse ikinci satıra `rdf:RDF` tanımlaması yapılmak zorundadır. İkinci, üçüncü ve dördüncü satırlarda RDF’de kullanılacak sözlüklerin URI’leri ve bunların yerine kullanılacak ad alanları görülmektedir. Örneğin RDF için ad alanları olarak RDF belirlenmiştir. Satır beşteki `rdf:Description` özelliği ile tanımlamaların ne hakkında yapılacağı belirtilmiştir. Burada hakkında tanımlama yapılacak kaynak URI, daha önce tanımlanmış bir kaynak olduğundan, `rdf:about` özelliği ile gösterilmiştir. Altıncı ve yedinci satırlarda kaynak hakkında bilgiler verilmiştir. Satır yedide, boş düğümün `rdf:nodeID="abc"` olarak tanımlandığı görülmektedir. Satır dokuz ve on iki arasında ise boş düğüm için yapılan özellik tanımlamaları görülmektedir. On üçüncü satırda RDF dokümanının sonlandırıldığı, `</rdf:RDF>` özelliği ile belirtilmiştir.

Bağlı Veri uygulamalarında RDF kullanımı; veri setleri, RDF sözlükleri ve ontolojiler arasında bağlantı kurma ve bu içeriği sorgulama açısından büyük önem taşımaktadır. RDFS, OWL gibi farklı ontoloji dilleri RDF ile birlikte kullanılabilir. Yapılandırılmamış, yarı yapılandırılmış ve yapılandırılmış veriler RDF ile modellenenebilir. Böylece Web ortamında RDF ile ifade edilmiş tüm kaynaklar birbiriyle bağlantılı hale gelir. RDF verilerinin farklı sözlükler kullanılarak, küresel ve eşsiz tanımlayıcılarla birlikte tanımlanması, herhangi bir kimsenin herhangi bir konuya atıf yapabilmesine olanak verir.

Çizelge 3.3’te RDF/XML olarak gösterilen örnek, RDF 1.0 ve RDF 1.1 versiyonlarına uygun olarak tasarlanmıştır. RDF 1.1; URI yerine IRI’lerin kullanılması, kaynak tanımlayıcılarını belirlemede RCF3986 standardına uyulması ve boş düğümlerin lokal referanslara sahip olması gerektiğini önermektedir. RDF 1.1 ile ayrıca, RDF/XML’den farklı yeni veri formatları önerilmiştir (bkz. Şekil 3.15). Bu yeni formatlardan bazıları çoklu çizgeleri sunma özelliğine sahiptir [80]. RDF/XML, Turtle ve Özniteliklerle Kaynak Tanımlama Çatkısı (Resource Description Framework in Attributes – RDFa) bağlı veri yayınlamada sıklıkla kullanılan RDF formatlarıdır.



Şekil 3.15 RDF 1.1 yeni model formatları [80]

3.3.1 RDF Veri Modeli Formatları

RDF farklı formatlarda düzenlenebilir. RDF'nin bir veri formatı değil; kaynakları özne, nesne ve yüklem formatında tanımlamak için bir veri modeli olduğunu hatırlamak önemlidir [14]. W3C tarafından RDF/XML, RDFa JSON-LD ve Turtle gibi RDF formatları standartlaştırılmıştır. Bunların dışında standart olmayan, fakat tüm ihtiyaçları karşılayan veri formatları da bulunmaktadır⁹.

3.3.1.1 RDF/XML

W3C standardı olan ve en yaygın kullanılan RDF formatıdır. İnsanlar tarafından okunup anlaşılması zordur [11]. RDF/XML formatında RDF hazırlamak için birçok farklı araç bulunmaktadır. Ayrıca, bazı coğrafi bilgi sistemleri ve veri tabanlarından RDF oluşturma metotları mevcuttur [17]. HTTP içerik uzlaşmasında MIME tipi, application/rdf+xml olarak belirlenmelidir. Çizelge 3.'teki örnek RDF/XML formatındandır¹⁰.

⁹ http://www.w3.org/standards/techs/rdf#w3c_all

¹⁰ <http://www.w3.org/TR/2014/REC-rdf-syntax-grammar-20140225/#section-grammar-summary>

3.3.1.2 RDFa

RDF, ayrı bir doküman olarak tutulmak yerine HTML dokümanların içine gömülebilir. RDFa¹¹ formatı, var olan HTML sayfasının RDF ile biçimlendirilmesine olanak tanır; var olan Web içeriğinin yapılandırılmış verileri içermesini sağlar. TopBraid Composer Word Press 2.7, DreamWeaver (x)HTML+RDFa eklentisi ve Drupal, RDFa üretilmesine olanak sağlamaktadır [17].

Çizelge 3.4 RDFa örneği [14]

```
1 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML+RDFa 1.0//EN" "http://www.w3.org/MarkUp/DTD/xhtml-rdfa-1.dtd">
2 <html xmlns="http://www.w3.org/1999/xhtml" xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
3   xmlns:foaf="http://xmlns.com/foaf/0.1/">
4 <head>
5 <meta http-equiv="Content-Type" content="application/xhtml+xml; charset=UTF-8"/>
6 <title>Profile Page for Dave Smith
7 </head>
8
9 <body>
10 <div about="http://biglynx.co.uk/people#dave-smith" typeof="foaf:Person">
11 <span property="foaf:name">Dave Smith
12 </div>
13 </body>
14
15 </html>
```

Çizelge 3.4'te örnek bir RDFa dokümanı verilmiştir. RDF/XML'de rdf:RDF tagları arasında yapılan tanımlamaların, bu örnekte html tagları arasında yapıldığına dikkat edilmelidir. RDFa kullanılarak Web'de Bağlı Veri yayımlanacağı zaman, verideki gerçek dünya nesneleri ve HTML+RDFa dokümanındaki tanımlamaların birbirine karışmamaları için, "about=" özelliğinin gerçek dünya nesnelerini ifade etmede kullanılması önerilmiştir. Birden fazla dokümanla uğraşmaktansa (RDF, HTML ,vd.), tek dokümanla yönetilebilir olduğundan, tercih edilen bir formattır [81].

3.3.1.3 Turtle

Turtle¹², insanlar tarafından anlaşılması en kolay formattır. Herhangi bir araç kullanmayıp doğrudan basit editörler ile RDF üretildiği durumlarda tercih edilir. Diğer

¹¹ <http://www.w3.org/TR/rdfa-primer/>

¹² <http://www.w3.org/TR/2014/REC-turtle-20140225/>

formatlara göre daha kısadır. RDF dokümanlarını sorgulamak için kullanılan SPARQL ve GeoSPARQL, Turtle formatını kullanarak oluşturulmuştur.

Çizelge 3.5'te Turtle formatında üretilmiş bir RDF örneği görülmektedir. Dördüncü satırda tanımlaması yapılacak nesne URI ile belirlenmiş, izleyen satırlarda bu nesne ile ilgili tanımlamalar yapılmıştır [14], [17], [82].

Çizelge 3.5 RDF Turtle örneği [14]

```
1 @prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .  
2 @prefix foaf: <http://xmlns.com/foaf/0.1/> .  
3  
4 <http://biglynx.co.uk/people/dave-smith>  
5 rdf:type foaf:Person ;  
6 foaf:name "Dave Smith" .
```

3.3.1.4 N-Triples

N-Triples¹³, Turtle formatının bir alt kümesidir. Ad alanı ve kısaltma gibi özellikler sağlamaz. Boyutu diğer formatlara göre daha büyüktür; çünkü tüm URI'ler her RDF ifadesinde tekrarlanarak belirtilir. Ancak, bu yaklaşım N-Triples formatına bir avantaj sağlar: Her N-Triples satırının bir anlamı vardır ve her satır ayrı ayrı bir anlam ifade eder [14], [17]. Çizelge 3.6'da örnek bir N-Triples formatındaki bir RDF örneği görülmektedir. Her satırın ayrı bir RDF ifadesi, bir RDF grafiği olduğu açıkça görülmektedir [83].

Çizelge 3.6 RDF N-Triples örneği [14]

```
1<http://biglynx.co.uk/people/dave-smith><http://www.w3.org/1999/02/22-rdf-syntax-  
ns#type><http://xmlns.com/foaf/0.1/Person> .  
2 <http://biglynx.co.uk/people/dave-smith><http://xmlns.com/foaf/0.1/name> "Dave Smith" .
```

3.3.1.5 JSON-LD

JSON-LD¹⁴, JSON formatında Bağlı Veri yayımlamamak için W3C tarafından geliştirilmiş bir sözdizim formatıdır. Bağlı Veri kullanarak Web tabanlı programlama, birlikte

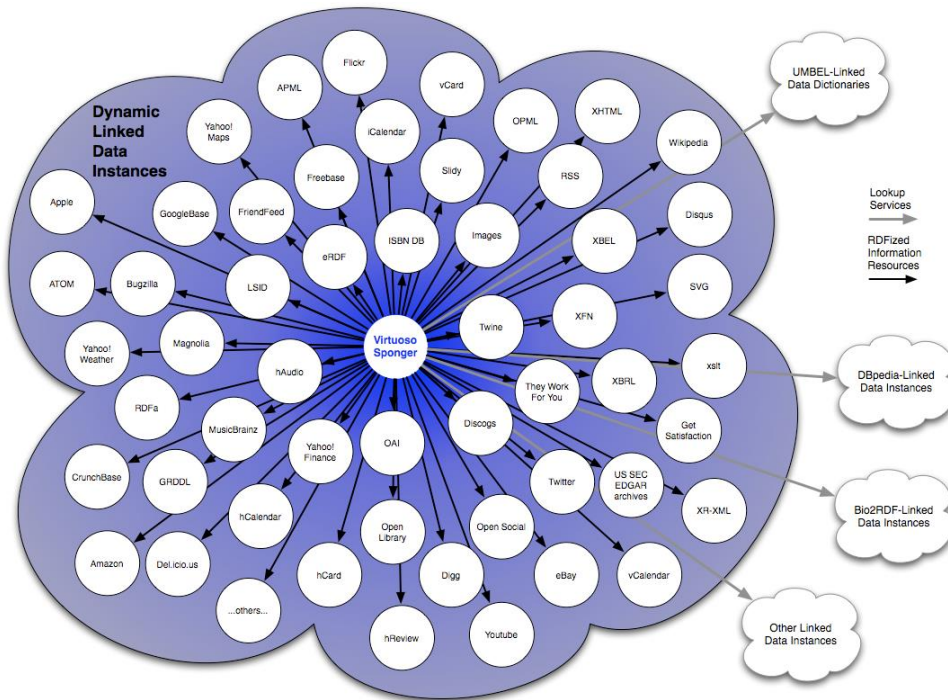
¹³ <http://www.w3.org/TR/2014/REC-n-triples-20140225/>

¹⁴ <http://www.w3.org/TR/json-ld/>

çalışabilir Web servisleri ve JSON tabanlı depolama sağlamak için geliştirilmiştir. JSON, formatında küçük değişiklikler yapılarak JSON-LD formatına çevrilebilmektedir. Birçok programlama dili JSON desteği sağladığından, kullanılması önerilmektedir [84].

3.3.2 RDF Oluşturma ve Doğrulama Araçları

RDF formatında olmayan veri kaynaklarını yukarıda belirtilen RDF formatlarına çeviren yazılımlar veya servisler bulunmaktadır. Bu araçlara genel olarak ‘RDFizers’ adı verilir. ‘RDFizers’, bir *Çıkart*, *Dönüştür*, *Yükle* (Extract, Transform, Load – ETL) servisi veya yazılımı olarak düşünülebilir. Bu servisler içinde en geniş olanaklar sunan, Virtuoso sunucu ile bağlantılı olarak RDF eşleştirme (mapping) ve RDF’leştirme olanakları sağlayan “Sponger Cartridges” isimli yazılımdır. Sponger tarafından RDF formatında ifade edilebilen formatlar Şekil 3.16’da görülmektedir [85], [86], [87].



Şekil 3.16 Virtuoso Sponger ile RDF’leştirilebilen veri tipleri [85]

Sponger dışında RDF’ye dönüşüm yapan birçok araç bulunmaktadır. Bu araçlar; CSV, excel dosyaları, elektronik postaları, Küresel Konumlama Sistemi (Global Positioning System – GPS) verileri, Yapılandırılmış Sorgu Dili (Structured Query Language – SQL) verileri, UML dokümanları, XML dokümanları ve daha birçok farklı formattaki dosyayı

RDF formatına dönüştürür. Google Refine¹⁵, Top Braid Composer¹⁶, D2RQ¹⁷ ve XLWrap¹⁸ bu araçlar arasında en sık kullanılanlardır. RDF formatına dönüşüm yapan araçların tam listesi W3C tarafından oluşturulmuştur¹⁹.

Üretilen RDF dokümanları online veya başka yazılımlar ile doğrulanabilmektedir. Jena tarafından oluşturulan RDF API²⁰ ve Sesame²¹, RDF dokümanlarını W3C RDF tanımlamalarına göre doğrulamakta ve sorgulayabilmektedir. W3C tarafından sağlanan doğrulama servisine <http://www.w3.org/RDF/Validator/> adresinden ulaşılabilir. Servis doğrudan yüklenen bir RDF dokümanını doğrulayabildiği gibi, Web’de bulunan RDF dosyalarını da sorgulayabilmektedir. RDF formatları arasındaki dönüşümler için de çeşitli online servisler ile gerçekleştirilebilir^{22,23,24}.

3.3.3 Coğrafi Veri ve RDF

Bir bileşeni konum olan her veri, coğrafi veri olarak kabul edilebilir. Coğrafi veri temel olarak; geometri, mereolojik ve topolojik ilişkiler ile metinsel öznitelik tanımlamalarına sahiptir. Topoğrafik haritalar, yer adları dizinleri, sayısal arazi modeli ve üç boyutlu modeller coğrafi verinin farklı sunum yöntemlerinden bazılarıdır. Bu yöntemler genellikle veriyi saklamak için klasik veri tabanı yaklaşımını kullanmaktadırlar. Ancak, son zamanlarda coğrafi verilerin modellenmesinde ve saklanmasında RDF veri modeli kullanılmaya başlanmıştır. Örneğin, Geonames ve OS tarafından yapılan çalışmalarda coğrafi veriler RDF sözlükleri ile modellenmiş, RDF formatında sunulmuştur.

¹⁵ <https://code.google.com/p/google-refine/>

¹⁶ <http://www.topquadrant.com/tools/IDE-topbraid-composer-maestro-edition/>

¹⁷ <http://d2rq.org/>

¹⁸ <http://xlwrap.sourceforge.net/>

¹⁹ <http://www.w3.org/wiki/ConverterToRdf/>

²⁰ http://jena.apache.org/tutorials/rdf_api.html

²¹ <http://rdf4j.org/>

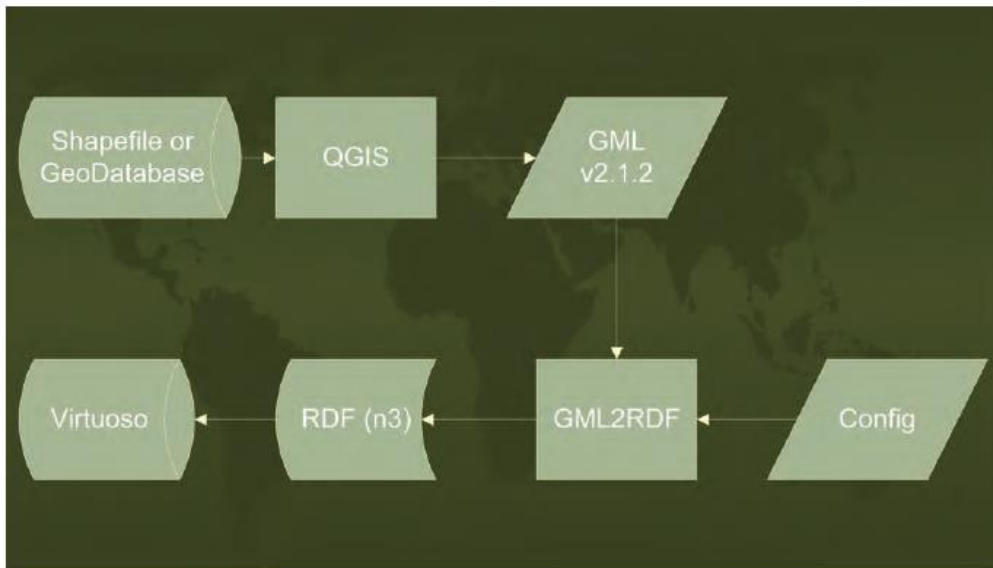
²² <http://www.ebusiness-unibw.org/tools/rdf2rdfa/>

²³ <http://rdf-translator.appspot.com/>

²⁴ <http://www.easyrdf.org/converter/>

Veri tabanları coğrafi veriyi saklamada bazı sorunlara neden olmaktadır. Örneğin, veri tabanında bir öznitelik sütünü açıldığında tüm coğrafi detaylar için o sütünün doldurulması gerekmektedir. Fakat RDF verileri üçlü ifadeleri olarak sakladığından, belirlenen özniteliklerin tüm coğrafi detaylar için tanımlanma zorunluluğu yoktur. Benzer şekilde, CBS’lerde zamansal faktörlerin tutulmasının zorluğu, RDF’de farklı URI’ler belirleyerek kolayca çözülmektedir. Örneğin, bir binanın koordinatlarının veya kullanımının değişmesi eski verileri kaybetmeden, yeni bir URI ile çözülebilir [17].

Coğrafi verilerin doğrudan RDF veri modelinde sunulması ve sorgulanabilmesi için GeoSPARQL standardı ortaya çıkmıştır. GeoSPARQL ile coğrafi koordinatlar GML, Bilinen Geometri Yazısı (Well-Known Text – WKT) gibi veri formatlarında RDF’de sunulabilir. Bunun yanında, Kuantum Coğrafi Bilgi Sistemi (Quantum Geographic Information System – QGIS) gibi bazı CBS yazılımları coğrafi veriyi GML olarak ifade etmektedir. GML olarak üretilen verinin RDF veri modeline dönüştürülmesi ise bazı dönüşüm araçları ile yapılabilmektedir. Birleşik Devletler Harita Kurumu (The National Map of the United States – USGS) bu amaç için GML2RDF adında bir yazılım oluşturmuştur. Şekil 3.17’de, yazılımın GML veri formatını, RDF veri modelinin bir formatı olan N3 formatına çevirdiği görülebilmektedir [88]. Bunun dışında birçok yazılım, farklı RDF sözlükleri ve ontolojilerle birlikte coğrafi bileşeni olan verileri RDF veri modeli olarak ifade edebilmektedir.



Şekil 3.17 GML'den RDF'ye dönüşüm [88]

Son yıllarda, INSPIRE kapsamında XSD, GML gibi formatlarla sunulan konumsal verilerin RDF formatına dönüştürülmesine ilişkin çalışmalar ivme kazanmıştır. UML ve XSD formatlarında üretilen veri temalarının, ontolojilerle ve RDF dokümanları ile ifade edilmesine ilişkin Yeniden Kullanılabilir INSPIRE Referans Platformu (A Reusable INSPIRE Reference Platform - ARE3NA) adlı proje kapsamında çeşitli kılavuzlar hazırlanmıştır [89].

3.4 Ontolojiler

RDF veri modeli kaynakları tanımlamak için, özne, yüklem ve nesne üçlüsünü kullanarak soyut bir veri modeli sağlar. Fakat RDF gerçek dünya nesneleri ve soyut kavramları tanımlamak ve bunların birbiriyle ilişkilerinin nasıl olduğu hakkında, alana özel tanımlamalar sunmaz. Herhangi bir alana özgü terimler ve bu terimler arasındaki ilişkiler; taksonomiler, RDF sözlükleri ve ontolojilerle sunulur [14]. Taksonomiler sadece parça-bütün ilişkisi kurabildiğinden, daha açık tanımlamalar yapabilmek için ontolojiler kullanılır. Ontoloji, 'paylaşılmış bir kavramsallaştırmanın formel ve açık bir spesifikasyonu'dur [18]. Ontolojiler; bilgi mühendisliği, yapay zeka, bilgisayar bilimleri, bilgi yönetimi, Anlamsal Web ve CBS ile ilgili uygulamalarda yaygın biçimde kullanılmaktadır [90]. Ontolojiler; kategorileri, kavramları, ilişkileri ve kuralları formel biçimde tanımlayarak, bilginin insanlar ve makineler tarafından paylaşılmasını sağlar; farklı sistemlerden gelen verilerin entegrasyonuna ilişkin problemleri minimize ederek birlikte çalışabilirliği olanaklı kılar [91]; verilerden ayrı olarak tanımlamalar yapıp, veriyi biçimlendirmeyi amaçlar.

Ontoloji kavramının doğuşundan bugüne farklı düzeylerde ontoloji çalışmaları yapılmıştır. İlk çalışmalarda tüm Web'i kapsayan üst düzey ontolojiler, bilgi tabanları geliştirilmiştir. "Top level" veya "Upper level" ontoloji olarak bilinen OpenCyc, Üst Düzey Birleştirilmiş Ontoloji²⁵ (Suggested Upper Merged Ontology – SUMO), Dilbilimsel ve Kavramsal Mühendislik için Tanımlayıcı Ontoloji²⁶ (Descriptive Ontology for Linguistic and Cognitive Engineering – DOLCE) gibi ontolojiler, çok genel olarak,

²⁵ <http://www.adampeace.org/OP/>

²⁶ <http://www.loa.istc.cnr.it/old/DOLCE.html>

kavramlar hakkında ansiklopedik bilgiler vermektedir. Ancak, kavramlar ve kavramlar arasındaki ilişki tanımlamalarını çok genel düzeyde yapan bu üst düzey ontolojiler, doğası gereği, daha düşük düzeylerdeki gereksinimleri karşılayamamaktadır. Daha açık bir anlatımla, üst düzey ontolojilerdeki kavramlar belirli alanlara (örn., arazi yönetimi, jeodezi) ilişkin kavramları ayrıntılı içermez. Bu nedenle, güncel ontoloji geliştirme çalışmaları daha çok alan (domain) ve uygulama ontolojilerine odaklanmıştır. Alan ontolojileri belirli bir alana ilişkin kavramlar ve kavramlar arasındaki ilişkileri ifade eden ontolojilerdir. Bu tez çalışması kapsamında üretilecek Türkiye İdari Birim Ontolojisi de farklı ontolojilerden yararlanarak geliştirilecek bir alan ontolojisidir.

3.4.1 Ontoloji Dilleri

Kavramlar ve kavramlar arasındaki ilişkileri farklı düzeylerde temsil eden, W3C tarafından standart olarak kabul edilmiş çeşitli ontoloji dilleri bulunmaktadır. İzleyen alt bölüm, ontoloji mühendisliğinde yaygın biçimde kullanılan SKOS, RDFS ve OWL gibi ontoloji dillerini özetlemektedir.

3.4.1.1 SKOS

SKOS; sınıflandırma şemaları, taksonomiler, ilişkisel kavramlar dizinler gibi basit bilgi organizasyon şemalarının temsili için geliştirilmiş bir dildir. Alan terimlerini sadece hiyerarşik (broader/narrower) ve bağlantısal (related) ilişkilerle organize eden kontrollü sözlüklerin temsiliinde tercih edilir. SKOS, hiyerarşik ve bağlantısal ilişkilerin dışındaki daha kapsamlı ilişkileri temsil edemez [19].

Çizelge 3.7 SKOS formatında bir ilişkisel kavramlar dizini – SKOS sunumu [92]

```
<skos:Concept rdf:about="http://www.cadastralvocabulary.org/CaLAtThe/AdministrativeUnit">
  <skos:prefLabel>Administrative unit</skos:prefLabel>
  <skos:scopeNote>GEMET theme</skos:scopeNote>
  <skos:definition>Units of administration, dividing areas where Member States have and/or exercise jurisdictional rights, for
  local, regional and national governance, separated by administrative boundaries (Source: GEMET).</skos:definition>
  <skos:changeNote>Established with version 1.0</skos:changeNote>
  <skos:inScheme rdf:resource="http://www.cadastralvocabulary.org/CaLAtThe"/>
  <skos:altLabel>Jurisdiction</skos:altLabel>
  <skos:altLabel>Spatial unit group</skos:altLabel>
  <skos:broader rdf:resource="http://www.cadastralvocabulary.org/CaLAtThe/AdministrativeFeature"/>
  <skos:broader rdf:resource="http://www.cadastralvocabulary.org/CaLAtThe/Boundary"/>
  <skos:narrower rdf:resource="http://www.cadastralvocabulary.org/CaLAtThe/NUTSRegion"/>
</skos:Concept>
```

Şekil 3.18’de kadastro ve arazi yönetimine ilişkin SKOS formatında geliştirilen kavramlar diziniyle [92] tanımlanmış ‘idari birimler’ teriminin HTML sayfası; Çizelge 3.7’de ise idari birimler teriminin SKOS ifadesi gösterilmektedir.

Administrative unit

Alternative label:

[Jurisdiction](#)
[Spatial unit group](#)

Broader terms:

[Administrative feature](#)
[Boundary](#)

Narrower terms:

[NUTS Region](#)

Related terms:

Definition:

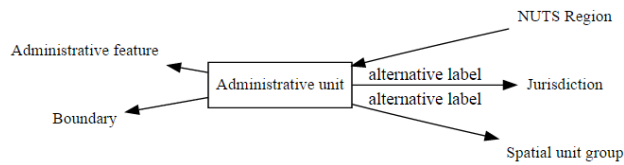
Units of administration, dividing areas where Member States have and/or exercise jurisdictional rights, for local, regional and national governance, separated by administrative boundaries (Source: GEMET).

Scope Note:

GEMET theme

Identifier:

<http://www.cadastralvocabulary.org/CaLaThe/AdministrativeUnit>



Şekil 3.18 SKOS formatında bir ilişkisel kavramlar dizini – HTML sunumu [92]

SKOS genellikle kavramlar ve kavramlar arasındaki hiyerarşik ilişkilerin temsiline odaklanırken; RDFS ve OWL, kavramlar arasındaki detaylı ilişkilerin temsiline olanak tanır [14].

3.4.1.2 RDFS

RDF Şema (Resource Description Framework Schema - RDFS), W3C tarafından 2004 yılında standart olarak kabul edilmiş ve 2014 yılında 1.1 versiyonu ile güncellenmiştir. RDFS, alan kaynaklarının sınıflarla (kavramlar) temsil edilmesine ve sınıflar arasında hiyerarşik ve hiyerarşik olmayan anlamsal ilişkilerin belirtilmesine olanak sağlar. RDFS’de önce alan kavramlarını ifade eden sınıflar oluşturulur. Ardından bu sınıflar arasındaki ilişkileri ifade eden ilişki türleri belirlenir ve standart RDFS özellikleri ile ilişkiler kurulur [20], [21]. RDFS tanımları Semantik Web ve Bağlı Veri yaklaşımı için

yeterli tanımlamalara sahiptir. Ancak, daha kapsamlı ilişki tanımlamaları için OWL tercih edilir [14], [17].

RDFS ile tanımlanan tüm varlıklar 'kaynak' (resource) olarak adlandırılır ve `rdfs:resource` sınıfının örneklemeleri olarak ifade edilirler. Kaynaklar genellikle URI'lerle tanımlanır. `rdfs:Class` özelliği, kaynakların RDFS sınıfları olarak ifade edilmesini sağlar. `rdfs:Literal`, `rdfs:DataType`, `rdf:LangString`, `rdf:HTML` ve `rdf:XMLLiteral`; RDFS'de kullanılacak veri değerlerini gösteren sınıflardır. RDFS'de kullanılan tüm özellikler `rdf:property` özelliğinin alt sınıfı ile tanımlanır. Bunlardan `rdfs:domain` ve `rdfs:range`, RDF sınıf örneklemelerinin alabileceği değerleri belirten niteleyicilerdir. Bu değerler başka bir sınıftan alınabileceği gibi, doğrudan veri tipi olarak tanımlanmış tam sayı, yazı gibi değerler de olabilir. `rdf:type` kaynağın hangi sınıfa ait olduğunu gösterir. `rdfs:label` kaynağa bir isim atamak, `rdfs:comment` ise kaynak hakkında bilgi vermek için kullanılabilecek bilgilendirme (annotation) özelliklerdir. RDFS'de hiyerarşik sınıflar ve özellikler tanımlanabilmektedir. Alt sınıfları tanımlamada `rdfs:subClassOf`, alt özellikleri tanımlamada `rdfs:subPropertyOf` özellikleri kullanılır. `rdf:Statement`, `rdf:subject`, `rdf:predicate`, `rdf:object` gibi özellikler de RDF ifadeleri hakkında bilgi sağlar. Bu özelliklerin yanında, RDF kaynaklarının diğer ontolojilerle bağlantılar kurulmasına olanak veren iki önemli özellik bulunmaktadır. Bunlardan birincisi, `rdfs:seeAlso` özelliğidir; RDF kaynağı ve başka kaynaklar ile bağlantı kurulmasını sağlar. Bu özellik, kaynak hakkında ilave bilgi sağlayacak başka bir kaynağın URI'sine yönlendirme yapar. İkincisi, `rdfs:isDefinedBy` özelliği ise kaynağın zaten başka bir şema tarafından tanımlandığını ifade etmek için kullanılır [20], [21].

Çizelge 3.8, bir RDFS örneğinin bir bölümünü göstermektedir. İlk yedi satırda şema içinde kullanılan diğer RDF şemaları ve ontolojiler, ad alanları ile birlikte verilmiştir. Dokuzuncu ve onuncu satırda RDF tipinin bir ontoloji olduğu belirtilerek, ontoloji adı tanımlanmıştır. On iki ve on beşinci satırlar arasında RDFS ontolojisi hakkında meta veriler verilmiştir. On yedi ve yirminci satırlar arasında 'Production' kaynağı ile ilgili bilgiler verilmiştir. Bu sınıfın bir RDFS sınıfı tipi olduğu, ismi ve sınıfla ilgili bir açıklama belirtilmiştir.

Çizelge 3.8 Bir RDFS örneği [14]

```
1 @prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
2 @prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
3 @prefix owl: <http://www.w3.org/2002/07/owl#> .
4 @prefix dcterms: <http://purl.org/dc/terms/> .
5 @prefix foaf: <http://xmlns.com/foaf/0.1/> .
6 @prefix cc: <http://creativecommons.org/ns#> .
7 @prefix prod: <http://biglynx.co.uk/vocab/productions#> .
8
9 <>
10   rdf:type owl:Ontology ;
11   rdfs:label "Big Lynx Productions Vocabulary" ;
12   dcterms:creator<http://biglynx.co.uk/people/nelly-jones> ;
13   dcterms:publisher<http://biglynx.co.uk/company.rdf#company> ;
14   dcterms:date "2010-10-31"^^xsd:date ;
15   cc:license<http://creativecommons.org/licenses/by-sa/3.0/> .
16
17 prod:Production
18   rdf:type rdfs:Class;
19   rdfs:label "Production";
20   rdfs:comment "the class of all productions".
```

3.4.1.3 OWL

OWL, W3C tarafından 2004 yılında standart olarak kabul edilen ve 2014'te ikinci versiyonu ile güncellenen yüksek ifade gücüne sahip bir ontoloji dilidir. OWL daha önce DAML+OIL olarak yine W3C tarafından standart olarak kabul edilen Web ontoloji dilinin devamı niteliğindedir.

OWL'in ilk versiyonunda OWL Full, OWL DL ve OWL Lite olmak üzere üç farklı OWL türü tanımlanmış; OWL 2 versiyonunda ise bu türler OWL Full ve OWL DL olmak üzere iki üst tanımlamaya yapılmıştır. Tüm OWL dilleri karmaşık bilgi yapılarını ifade edebilmek için Birinci Derece Mantığa (First Order Logic – DL) dayanarak üretilmiştir. OWL'nin RDFS'den farkı, daha karmaşık ifadelerin temsiline olanak vermesi ve belirtilmeyen ilişkilerin DL ile çıkarımlanabilmesidir (reasoning).

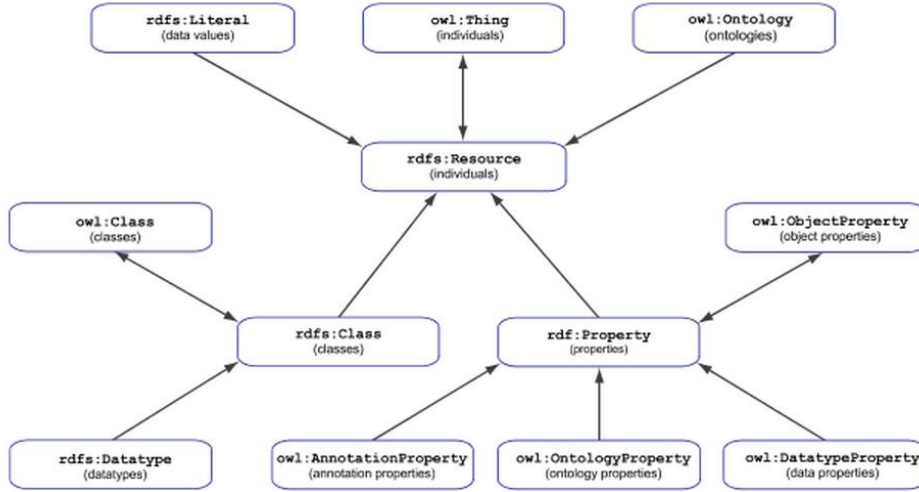
OWL farklı sözdizim formatlarında (syntax) üretilebilir; RDF/XML, OWL/XML, Manchester, Rabbit bunlardan bazılarıdır. Bu formatlar içinde, Rabbit düz yazıya en yakın olanıdır: İfadeler cümleler şeklinde temsil edildiğinden okuması ve anlaşılması çok kolaydır. W3C tarafından standart olarak kabul edilmiş Manchester formatı da insanlar tarafından okunması görece kolay olan bir formattır. XML dilinde gereğinden fazla tanımlama ile temsil edilen ifadeler Manchester formatıyla daha basit biçimde temsil edilir. İfadelerin karışık olduğu, insanlar tarafından kolay anlaşılmadığı ve makinelerin okurken (parse) zorlandığı ifade edilse de, günümüzde en yaygın olarak

kullanılan formatlar: RDF/XML ve OWL/XML'dir. Çoğu ontoloji geliştirme aracı da bu formatları desteklemektedir. Bu formatlar XQuery gibi XML araçları tarafından kullanılabilmekte, sorgulanabilmektedir [93], [94], [95], [96].

OWL, tanımlamalarını üç ana yapı ile biçimlendirir. Bunlar; sınıflar, ilişkiler (properties, slots, relations) ve örneklemelerdir (individuals, instances). OWL sınıfları, örneklemeleri içeren setler olarak açıklanabilir (Örneğin, bir kedi sınıfının tüm kedi örneklemelerini içermesi). OWL'de örneklemeler birden fazla sınıfa ait olabilir, yani bir örneklem birden fazla sınıf içinde yer alabilir. Sınıflar arasında hiyerarşik ve hiyerarşik olmayan ilişkiler tanımlanabilir ve bu ilişkilere geçişkenlik, simetriklik gibi fonksiyonel özellikler atanabilir.

OWL semantiğinde RDF ve RDFS tanımlamaları da yer almaktadır. Şekil 3.19'da OWL'deki sınıf yapısı ve bunun RDFS ile ilişkisi gösterilmektedir. RDFS'de kaynaklar rdfs:Resource sınıfı ile tanımlanırken, OWL'de bu tanımlama için; owl:Ontology, rdfs:Literal ve owl:Thing sınıfları kullanılır. owl:Thing sınıfı, OWL ontolojilerinde her zaman süper sınıf olarak yer alır. Tüm tanımlamlar owl:Thing sınıfının alt sınıfı olarak tanımlanır.

RDFS'de ilişki türlerinin tanımlanmasında kullanılan rdf:Property sınıfı, OWL'de karmaşık ilişki türlerinin ifade edilebilmesi için detaylandırılmıştır: Nesne ilişkileri (owl:ObjectProperty), veri tipi ilişkileri (owl:DatatypeProperty), açıklama ilişkileri (owl:annotationProperty) ve ontoloji ilişkileri (owl:OntologyProperty). rdfs:Class sınıfının özellikleri de genişletilerek owl:Class sınıfı olarak OWL'de yerini almıştır [22], [95], [96].



Şekil 3.19 OWL 2 semantik hiyerarşisi [96]

OWL dilinin zenginliği, sınıflar arasında birçok farklı ilişki kurabilme yeteneğinden kaynaklanmaktadır. Bunlardan `owl:disjointClasses` ilişkisi, sınıfları ayırmak ve ayrılan sınıfların çokluk değerlerini ifade etmek için kullanılabilir. Bu ilişki ile sınıflar birbirinden bağımsız hale getirilir ve bir örneklemenin, birden fazla sınıfın örneklemesi olması önlenir. Yani, bir sınıf içinde tanımlanan bir örneklemenin, bir başka sınıfın örneklemesi olması istenmiyorsa sınıflar arasında bu ilişki kurulmalıdır [17], [97]. OWL'de sınıflar arası hiyerarşi kurulabilmektedir. Bunun için `rdfs:subClassOf` ilişkisi kullanılabilir. `owl:imports` ilişkisi, başka ontolojilerin sınıflarının ve ilişkilerinin kullanılmasına olanak tanır. Bir diğer ontolojinin tüm tanımlamaları, oluşturulacak olan ontolojide kullanılmak isteniyorsa, bu ilişki kullanılabilir. Bu ilişkinin kullanılması, o ad alanındaki tüm tanımlamaların onaylandığını ifade eder. Bunun yanında yine başka ontolojilerde kullanılan sınıflar arasında, `owl:equivalentClass` ilişkisi ile semantik eşitlik sağlanır. Bu ilişki iki sınıfı birbirine tamamen eşitlemekte, sınıfların aynı olduğunu ifade etmektedir. Sınıf özelliklerin birbirinden ayrı ve farklı olduğunu ifade etmek içinse `owl:disjointwith` ilişkisi kullanılabilir. OWL 2 versiyonunda sınıflar birbirinin bileşimi olarak da ifade edilebilmektedir. `owl:disjointUnion` ilişkisi bir sınıfın, birbirinden bağımsız tüm alt sınıflarını gösteren ilişkidir. İki sınıfın birbirinden farklı olduğunu belirtmek için `owl:differentFrom` ilişkisi kullanılabilir [17], [22], [95].

Sınıflara atanan değerleri sınırlayan aksiyomlar, değer kısıtlamaları olarak adlandırılır. Bunlar, owl:someValuesFrom - varlıksal niteleyiciler (existential qualifier) ve owl:allValuesFrom - evrensel niteleyicilerdir (universal qualifier). Varlıksal niteleyiciler, sınıf ve ilişki arasında 'en az' bir (some) ilişkisi tanımlamasını yapar. Örneğin, 'Her bar en az bir bira türü satmak zorundadır.' tanımlamasını OWL'de ifade ederken, varlıksal niteleyicinin kullanılması gerekmektedir. Evrensel niteleyiciler, sınıf ve ilişki arasında 'sadece' (only) tanımlaması yaparken kullanılır. Sınıfın ilişki üzerindeki değerlerini sadece belirlenen sınıftan alabileceğini belirtir. Örneğin, 'Vejeteryan pizza malzemelerini sadece sebze sınıfından alır.' ifadesini OWL'de tanımlamak için evrensel niteleyiciler kullanılır. Sınıflar ve ilişkiler arasında kurulan bu ilişkiler kısıtlama ilişkileridir ve owl:Restriction sınıfı ile ifade edilir. Kısıtlamalar, OWL'de adlandırılmamış, tanımlanmamış sınıflar olarak yer alırlar [17], [97], [98].

OWL'de farklı ilişki türleri bulunmaktadır. Nesne ilişkileri bir örneklemeyi diğerine bağlar. Veri tipi ilişkileri bir örneklemeyi yazı, sayı gibi bir veri tipine bağlar. Açıklama ilişkileri sınıflar, ilişkiler ve örneklemelerle ilgili meta veri eklemede kullanılmaktadır. Eğer tanım veya değer kümesi birer sınıfsa nesne ilişkisi, eğer kümelerden biri XSD veri tipi ise veri tipi ilişkisi olarak tanımlanır. OWL'de ve genellikle, ilişki tanımlamaları küçük harfle başlayıp, daha sonraki her kelimenin baş harfi büyük harfle yazılırken, sınıf tanımlamalarında her kelimenin ilk harfi büyük olarak yazılır.

İlişkiler de sınıflar gibi hiyerarşik olarak tanımlanabilmektedir. İlişki hiyerarşilerinde tanım kümesi ve değer kümesi kısıtlamaları, alt ilişkilere aktarılmaktadır. Fakat ilişkilere tanımlanabilen simetriklik, geçişkenlik gibi özellikler, alt ilişkilere aktarılmamaktadır. OWL'nin RDFS ontolojilerinden bir diğer farkı da ilişkilere fonksiyonel özellikler tanımlanabilmesidir. Bu ilişkilerden ilki terslik (owl:inverseOf) ilişkisidir. Örneğin, 'ailesidir', 'çocuğudur' ilişkileri birbirine ters ilişki olarak tanımlanabilir. Bir örneklemekten diğerine en fazla bir ilişki tanımlanabiliyorsa o ilişki fonksiyonel bir ilişkidir (owl:FunctionalProperty). Fonksiyonel ilişki çokta bir değerleri ifade eder. 'annesidir' ilişkisi buna örnek olarak gösterilebilir. Ters fonksiyonel ilişkiler ise bire çok ilişkileri tanımlar (owl:InverseFunctionalProperty). Vatandaşlık kimlik numaraları ters fonksiyonel ilişkilere örnek olabilir. Geçişkenlik ilişkilerinde eğer bir ilişki a örneklemesini b'ye, b'yi de c örneklemesine bağlıyorsa a ve c örneklemelerinin de

birbirine aynı ilişki ile bağlı olduğunu gösterir (owl:TransitiveProperty). ‘akrabasıdır’ ilişkisi geçişken bir ilişkidir. Simetrik ilişkisinde a örnekleme b’ye, b örnekleme de aynı şekilde a’ya bağlanabilir (owl:SymmetricProperty). ‘kardeşidir’ ilişkisi simetrik bir ilişkiye örnek olarak verilebilir. Simetrik olmayan ilişkiler de asimetric olarak tanımlanabilir. ‘çocuğudur’ ilişkisi asimetric bir ilişkidir. Refleksif ilişkilerde bir ilişki kendisiyle de ilişkilendirilebilir. Yansıyan ilişki olarak da adlandırılabilen bu ilişkiye ‘tanımak’ ilişkisi örnek olarak verilebilir. Bu ilişkinin tersi olan irrefleksif ilişkiye ise ‘annesidir’ ilişkisi örnek olarak verilebilir. OWL’de ilişkilere özellikler atanırken dikkat edilmesi gereken kurallar vardır. Örneğin, bir ilişki geçişken ise o ilişkinin tersi de geçişkendir veya bir ilişki geçişken ise kesinlikle fonksiyonel olamaz.

OWL, sınıfların olduğu gibi, farklı ontolojilerde tanımlanan ilişkilerin de birbirine eşitlenmesine olanak tanımaktadır. owl:equivalentProperty ilişkisi ile farklı ontolojilerdeki ilişkiler birbirine eşitlenebilmektedir. owl:sameAs ilişkisi de farklı ontolojilerde bulunan ilişkilerin tamamen birbirinin aynısı olduğunu ifade etmek için kullanılabilir. OWL’de ilişkilere nicelik sınırlayıcıları (cardinality, qualified) da atanabilmektedir. OWL ilişkilerin alabileceği minimum örnekleme belirtmek için owl:minCardinality, en fazla örnekleme belirtmek için owl:maxCardinality ve kesin değerleri belirtmek için ‘exact’ tanımlamaları kullanılır. owl:Restriction sınıfı tanım kümesi, değer kümesi olarak da xsd:nonNegativeInteger seçilerek, çokluk tanımlamaları tam sayı olarak yapılabilir. owl:onProperty ilişkisi, hangi ilişki üzerinde kısıtlamalar yapıldığını göstermek için kullanılmaktadır. Bunun yanında ilişkiler bir veri tipi ilişkisi ile de ilişkilendirilebilir. Burada XSD veri tipleri veya RDF literalleri kullanılır. Böyle ilişkilerde büyüktür, küçüktür matematiksel tanımlamaları da kullanılabilir [95], [98].

OWL, ilişkiler arasında kesişim ve birleşim tanımlamaları yapabilmektedir. owl:unionOf ilişkisi bir sınıfı bir liste olarak ifade eder. owl:unionOf ilişkisi ‘veya’ anlamına gelir. Sunulan listeden biri veya diğeri seçilebilir anlamında kullanılır. owl:intersectionOf ilişkisi ise ‘ve’ anlamına gelmektedir. Sunulan listeden biri, ikisi veya ikisi birden listeden seçilebilir, anlamındadır. Eğer owl:oneOf ilişkisi kullanılırsa sıralanan (enumerated) örnekleme birinin seçilmesi gerekmektedir [99].

OWL'de, açıklamalar ve ontoloji hakkındaki meta verileri (annotation) sunabilmek için birtakım ilişkiler kullanılmaktadır. Bunlar içinde; `rdfs:label` etiket bilgilerini, `rdfs:comment` ontoloji hakkındaki bilgileri, `owl:versionInfo` ontolojinin versiyonunu, `rdfs:seeAlso` ek bilgileri, `rdfs:isDefinedBy` ilgili tanımlamanın başka bir ontolojide yapıldığını sunmak için kullanılır. OWL 2 ile birlikte `owl:backwardCompatibleWith`, `owl:priorVersion`, `owl:versionInfo` ve `owl:incompatibleWith` gibi meta veri ilişkileri kullanılmaya başlanmıştır. Bunların yanında, başka RDF sözlükleri ve ontolojiler kullanılarak daha geniş kapsamlı meta veri sağlama imkanı bulunmaktadır [100].

Sonraki bölümde, yukarıda kısaca açıklanan ontoloji dilleri ile ontoloji geliştirilmesine yönelik metodolojiler anlatılacaktır.

3.4.2 Ontoloji Geliştirme Metodolojileri

Ontoloji kavramı bilgisayar biliminde kullanılmaya başlandıktan sonra birçok ontoloji geliştirme metodolojisi tasarlanmıştır. Bunlardan en çok kabul göreni METHONTOLOGY'dir [101]. Hart ve Dolbear (2012), kabul görmüş metodolojileri incelemiş ve ontoloji mühendisliği ilkeleri doğrultusunda, şu adımların uygulanması önerilmiştir [17]:

- 1) Amaç ve kapsam: Ontolojinin ne amaca hizmet edeceği ve sınırlarının ne olacağı açıkça belirlenmelidir. Bu, ontoloji dilinin seçiminde etkin rol oynar. Kompleks ontolojilerde RDFS yerine, OWL dili seçilmelidir.
- 2) Kullanım durumları ve tamlik testleri: Ontolojide kullanılacak kapsamın kullanım durumu incelenmelidir. Tamlik testlerinde ise ontolojinin amaç ve kapsamını tamamen sağlayıp sağlamadığı incelenir ve sorgulama gibi test çalışmalarıyla bunun kontrolünü sağlanır.
- 3) Terimler listesi: Ontoloji geliştirmeye başlamadan önce, ontoloji geliştirilecek alana ait bir terimler listesi (lexicon) hazırlanır. Bu aşamada, belirlenen kapsamla ilgili tüm dokümanlar incelenmelidir.
- 4) Terimler sözlüğü: Terimler listesine eklenen tüm terimlerin organize edildiği ve tanımlandığı aşamadır. Kavram hiyerarşisi oluşturulur. Kavramsallaştırma aşamasında buradan elde edilen veriler kullanılır.

- 5) Kavramsallaştırma: Doğal dilde oluşturulan terimlerin, kavramların formel bir ontoloji dili ile ifade edildiği aşamadır. Ontolojide ilk önce sınıflar tanımlanır. Sınıfları tanımlama aşamasında daha önceden bu kapsamında hazırlanan ontoloji sınıflarının kullanılıp kullanılmayacağı kararlaştırılır. Ontoloji kapsamı ile ilgili daha önceden yapılan çalışmalar incelenir. Üst seviye (süper) sınıflar belirlenir. Daha sonra bu sınıflara alt sınıflar atanır. Bu aşamada, ikili (binary) veya çoklu (n-ary) tasarım modeli ile ontoloji sınıf tanımlamaları yapılmaya başlanır. İlişki tanımlamaları yapılır. İlişkilerin özellikleri belirlenir ve sınıf tanımlamaları şekillendirilir. Ontoloji geliştirme modeline uymayan durumlar çözülmeye çalışılır. En son aşamada, oluşturulan bu sınıflar (describing), tanımlanmış (defining) sınıflara çevrilir. Tanımlanmış sınıflar otomatik olarak mantıksal çıkarım yapabilecek duruma gelmiş sınıflardır. Sınıflar gerekli ve yeterli koşulları tanımlamalarında sağlıyorsa ontoloji tanımlamaları tamamlanmış olmaktadır.

Ontoloji dillerinden birini seçerek ve seçilen bir metodolojiyle ontoloji tasarlamak için çeşitli araç ve uygulama geliştirilmiştir. Bir sonraki bölümde bu araçlar ve uygulamalar anlatılmaya çalışılacaktır.

3.4.3 Ontoloji Geliştirme Araçları

Herhangi bir dilde ontoloji geliştirmek için o dilin tüm tanımlamalarını iyi bilip uygun durumlarda kullanmak gerekir. Ancak OWL gibi karmaşık bir dilin herhangi bir sözdizim formatında bunu yapabilmek, pratikte oldukça güçtür. Bu nedenle ontoloji geliştirme araçlarına gereksinim duyulur. Bu araçlar ontoloji oluşturma işlevlerinin yanında, sağladıkları mantıksal çıkarsama araçlarıyla da geliştirilen ontolojileri denetleme olanağı verir.

Ontoloji geliştirme araçlarından en yaygın olarak kullanılanı, Protégé açık kaynak kodlu ontoloji editörüdür. Standford Üniversitesi tarafından geliştirilen bu editör, tüm OWL tanımlamalarını içermektedir. Editörün kullanımı kolay arayüzü, OWL dilinin tüm tanımlamalarının kolayca yapılmasını sağlar. Editör, 4.x versiyonlarından başlayarak RDF/XML, OWL/XML ve Manchester formatlarında ontoloji tasarımına olanak vermektedir. Protégé, Semantik Web Kural Dili (Semantic Web Rule Language – SWRL)

eklentisi sağlamaktadır. Ayrıca, Hermit²⁷, FaCT++²⁸, Pellet²⁹, RacerPro³⁰ ve TrOWL³¹ gibi çıkarsama eklentilerinin kullanılmasını sağlayarak ontolojilerin denetlenmesini sağlar.

Yaygın kullanılan bir diğer ontoloji geliştirme editörü ise Neologism'dir. Neologism açık kaynak kodlu olarak ve Drupal platformunda PHP formatında geliştirilmiştir. RDF sınıfları ve ilişkileri oluşturarak, basit sözlükler oluşturmak için kullanılmaktadır. [102].

NeOn Toolkit, NeOn projesi kapsamında Eclipse üzerinde geliştirilen, açık kaynak kodlu bir ontoloji geliştirme aracıdır. Birçok farklı eklenti sunmaktadır [103]. Bir başka ontoloji editörü online ve bulut tabanlı Knoodl'dur. Ontoloji oluşturma OWL dilinin, bilgi tabanı oluştururken RDF dilinin kullanılmasına olanak tanımaktadır. Ayrıca kullanıcıların kendi ontolojilerini, bilgi tabanlarını ve Knoodl tarafından sağlanan Java servis tabanlı veya SPARQL tabanlı arayüzlerini kullanarak, uygulama geliştirmelerine olanak sağlamaktadır [104]. TopBraid Composer bir başka ontoloji geliştirme editörüdür. Hemen bütün Semantik Web teknolojilerini bünyesinde barındıran bu yazılım, ticari bir yazılımdır. Veri dönüşümünden Web'de veri yayımlamaya, yayımlanan verileri SPARQL ile sorgulamaya kadar birçok Semantik Web uygulamasına olanak sağlar.

3.4.4 Varolan RDF Sözlükleri

Herhangi bir alana ait bilgileri düzenlemek için oluşturulan şemalar, RDF sözlüğü veya ontoloji olarak adlandırılır. Birçoğu RDF, RDFS veya OWL ile geliştirilmiştir. Ontoloji mühendisliği, herhangi bir alana ilişkin bir RDF sözlüğü geliştirirken, daha önce oluşturulan sözlüklere bakılmasını ve var olan sözlüklerin ihtiyacı karşılaması durumunda yeni bir sözlük oluşturulmamasını önerir. Ancak, bir veri seti tanımlarken yeni terimlerin geliştirilmesine ihtiyaç duyulan durumlar da olabilir. Böyle durumlarda, yeni eklenen terimler, iyi yapılandırılmış sözlüklerin ilgili terimleriyle eşleştirilmelidir.

²⁷ <http://hermit-reasoner.com/>

²⁸ <http://owl.man.ac.uk/factplusplus/>

²⁹ <http://clarkparsia.com/pellet/>

³⁰ <http://franz.com/agraph/racer/>

³¹ <http://trowl.eu/>

Kullanılabilecek sözlükleri seçerken, sözlüğün; yaygın olarak kullanılıp kullanılmadığına, terimleri ifade edebilirlik seviyesinin veri setini yeterince kapsayıp kapsamadığına ve sürdürülebilir olup olmadığına bakılmalıdır [14]. 2011 yılında güncellenen verilere göre Bağlı Veri uygulamalarında kullanılan sözlüklerin istatistiksel dağılımı <http://lod-cloud.net/state/#terms> adresinde sunulmuştur. Eklenerek yeni terimlerin sonraki uygulamalarda kullanımını etkinleştirmek için kontrol edilebilir bir ad alanıyla yayımlanması, terimlerin rdfs:label ve rdfs:comment gibi özelliklerle açıklanması ve terimleri ifade etmede kullanılan URI'lerin yeniden referanslandırılabilir olmasına özen gösterilmelidir. Ayrıca, gerekli olmayan durumlarda rdfs:domain, rdfs:range gibi tanımlamaların ve yersiz aksiyomların, RDF sözlüğünü şişirmesine izin verilmemelidir [14]. Günümüzde farklı alanlarda üretilmiş birçok RDF sözlüğü ve ontoloji bulunmaktadır³². Bu sözlükler içinde yaygın biçimde kullanılanlar, kullanım alanları ile birlikte aşağıdaki açıklanmıştır:

- **DC (Dublin Core) Sözlüğü³³:** Dublin Temel Meta Veri Girişimi Dublin (Dublin Core Metadata Initiative – DCMI) tarafından hazırlanmış bir meta veri sözlüğüdür. RDF örneklemelerini başlık, oluşturan kişi, oluşturma tarihi, format, dil gibi temel meta verilerle zenginleştirmek amacıyla kullanılır.
- **FOAF (Friend-of-a-Friend) Sözlüğü³⁴:** İnsanları, insanların birbiriyle ilişkilerini, faaliyetlerini ve nesnelerle ilişkilerini ifade etmek için oluşturulmuş bir sözlüktür. Bir kişinin adı, yaşı, e-mail adresi ile kişiyi ilgilendiren dokümanlarla ilişkiler sağlayabilecek özellikleri içermektedir.
- **CC (Creative Commons) Sözlüğü³⁵:** Hazırlanan RDF örneklemelerinin telif haklarını açıklamak için hazırlanmış, farklı lisanslama formatları sunan bir RDF sözlüğüdür.

³² <http://lov.okfn.org/dataset/lov/>

³³ <http://dublincore.org/documents/dcmi-terms/>

³⁴ <http://xmlns.com/foaf/spec/>

³⁵ <http://creativecommons.org/ns#>

- **SKOS Sözlüğü**³⁶: Kavramları, kavramların birbiriyle ilişkisini tanımlamak üzere hazırlanmış bir veri sözlüğüdür.
- **vCard Sözlüğü**³⁷: İnsanların yaş, cinsiyet gibi niteliklerini ve telefon, e-posta gibi bilgilerini ifade etmek için hazırlanan bir RDF sözlüğüdür.
- **XSD Sözlüğü**³⁸: RDF dokümanlarında nesnelerin alabileceği literal, URI gibi değerleri tanımlayan sözlüktür.

Tüm şemalarda kullanılabilecek nitelikte terimler içeren bu sözlüklerin dışında, belirli bir alana ait farklı RDF sözlükleri de bulunmaktadır. Örneğin, coğrafi verileri ifade etmek amacıyla sıklıkla kullanılan sözlükler şunlardır:

- **Basic Geo (WGS84) Sözlüğü**³⁹: Coğrafi koordinatları ve yükseklik bilgisini WGS84 sisteminde ifade etmek için oluşturulmuş bir sözlüktür.
- **GML Sözlüğü**⁴⁰: OGC tarafından hazırlanan ve GML’de kullanılabilecek tüm geometri türlerini içeren sözlüktür.
- **Basitleştirilmiş Coğrafi Detay Geometrisi Sözlüğü**⁴¹: OGC tarafından tanımlanan temel geometri tiplerini içeren sözlüktür.
- **NeoGeo Geometri Ontolojisi**⁴²: Coğrafi bölgeleri ifade etmek için geometri tanımlamalarını içeren bir sözlüktür.
- **NeoGeo Konumsal İlişkiler Ontolojisi**⁴³: Coğrafi detaylar arasındaki konumsal ilişkileri tanımlayan bir ontolojidir.
- **ISO Konumsal Şema Ontolojisi**⁴⁴: ISO 19107:2003 Konumsal Şemasının bir kısmının tanımlamalarını içeren ontolojidir. Geometri tipi tanımlamaları içerir.

³⁶ <http://www.w3.org/2009/08/skos-reference/skos.html#>

³⁷ <http://www.w3.org/TR/2014/NOTE-vcards-rdf-20140522/>

³⁸ <http://www.w3.org/2001/XMLSchema#>

³⁹ <http://www.w3.org/2003/01/geo/>

⁴⁰ http://schemas.opengis.net/gml/3.2.1/gml_32_geometries.rdf

⁴¹ <http://www.opengis.net/ont/sf#>

⁴² <http://geovocab.org/geometry#>

⁴³ <http://geovocab.org/spatial#>

- **ISO Coğrafi Detay Ontolojisi⁴⁵**: ISO 19109 Genel Coğrafi Detay Modelinin bir kısmının ontolojisidir.
- **Geonames Ontolojisi⁴⁶**: Coğrafi detay sınırlarını sınıflandıran bir ontolojidir. Ad, yüzölçüm gibi öznitelikler içerir.
- **OS Konumsal İlişkiler Ontolojisi⁴⁷**: Egenhover, RCC8 ve GeoSPARQL tanımlamalarını içeren bir konumsal ilişkiler ontolojisidir.
- **OS Geometri Ontolojisi⁴⁸**: GeoSPARQL tanımlamalarına uygun olarak basit geometri sunumlarını ifade etmek için hazırlanmış bir ontolojidir.
- **GeoSPARQL Ontolojisi⁴⁹**: OGC tarafından hazırlanan, konumsal veri setlerini SPARQL ile sorgulamak için birtakım geometrik ve topolojik tanımlamalar içeren ontolojidir.

Bağlı Veri yayımlama ilkelerinin üçüncüsünde, verilerin RDF ve SPARQL gibi standartlara uygun hazırlanması gerektiği ifade edilmektedir. İzleyen bölüm, SPARQL ve GeoSPARQL standartlarına değinmektedir.

3.5 SPARQL ve GeoSPARQL

3.5.1 SPARQL

SPARQL, Web’de RDF formatında sunulan veri setlerini sorgulamak için W3C tarafından oluşturulmuş bir sorgulama dili ve protokoldür. Web ve Semantik Web kavramlarını geliştiren Tim Berners-Lee, *‘Semantik Web’i SPARQL olmadan kullanmak veri tabanlarını SQL olmadan kullanmaya benzer*’ diyerek, SPARQL’in Semantik Web’deki önemine vurgu yapmıştır. SPARQL, RDF Veri Erişimi Çalışma Grubu (RDF Data Access Working Group – DAWG) tarafından oluşturulmuş ve 2008 yılında W3C tarafından bir

⁴⁴ <http://def.seegrid.csiro.au/isotc211/iso19107/2003/geometry#>

⁴⁵ <http://def.seegrid.csiro.au/isotc211/iso19109/2005/feature#>

⁴⁶ <http://www.geonames.org/ontology#>

⁴⁷ <http://data.ordnancesurvey.co.uk/ontology/spatialrelations/>

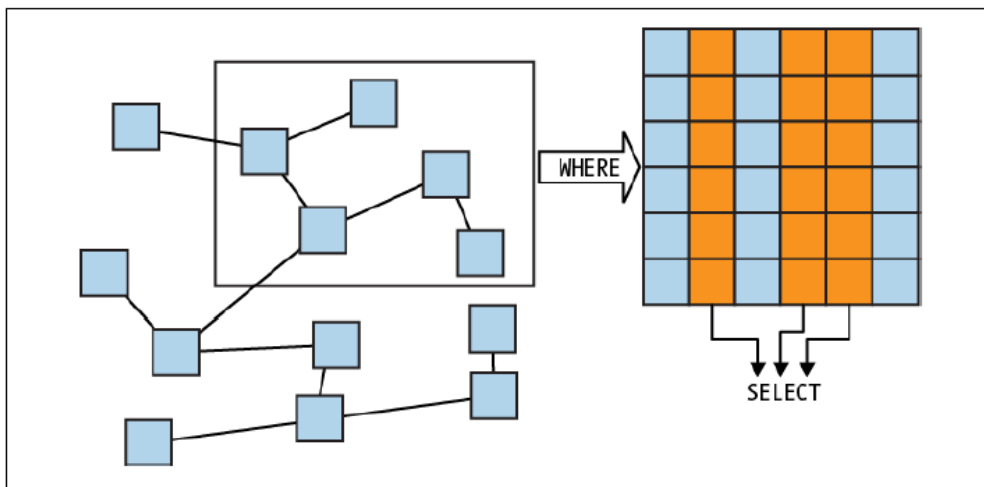
⁴⁸ <http://data.ordnancesurvey.co.uk/ontology/geometry/>

⁴⁹ <http://www.opengis.net/ont/geosparql#>

standart haline getirilmiştir. 2013 yılında ise güncellemelerle birlikte SPARQL 1.1 versiyonu yayımlanmıştır.

SPARQL, RDF sorgulama dilini destekleyen farklı sorgulama araçları bulunmaktadır. Bunlardan biri, Apache Jena tarafından Semantik Web ve Bağlı Veri uygulamalarında kullanılmak üzere geliştirilmiş, Jena için bir SPARQL İşlemcisi'dir (A SPARQL Processor for Jena – ARQ) [105]. ARQ, SPARQL 1.1 versiyonunun birçok tanımlamasını içerir ve birleştirilmiş, karmaşık sorgulamalara olanak sağlar. ARQ, SPARQL Endpoint (Sorgulama Noktası) olarak yayımlanmış farklı kaynaklara uzaktan erişim imkanı da sağlamaktadır. SPARQL sorgularına olanak tanıyan bir diğer yazılım ise Jena tarafından geliştirilen Fuseki'dir [106]. Fuseki bir SPARQL sunucusudur ve HTTP üzerinden RDF verilerini sunar. HTTP üzerinden 'put', 'get' 'query' ve 'update' yapabilmektedir. SPARQL sorgulamalarına olanak tanır. Ayrıca, verileri SPARQL Endpoint olarak da sunulabilmektedir. Fuseki, Jena tarafından üretilen bir RDF deposu (triple store) olan TDB'nin üzerine kurulmuştur. Bunlar dışında en çok kullanılan araçlardan biri Virtuoso'dur. Virtuoso da, Fuseki gibi SPARQL Endpoint sağlayabilmektedir ve ayrıca bir RDF deposu olarak da kullanılmaktadır [85], [86].

SPARQL, Web'de verilerini RDF olarak sunmuş tek bir kaynağı veya birden fazla kaynağı aynı anda sorgulayabilmektedir. Şekil 3.20'de görüldüğü gibi sorgulamalar genellikle SELECT, WHERE ve FROM tanımlamaları üzerinden oluşturulur. SELECT sorgulanmak istenen verileri, WHERE ve FROM ise bu verilerin hangi veri setlerinden seçileceğini belirlemek için kullanılmaktadır [24], [25].



Şekil 3.20 SPARQL ile Örnek RDF sorgulaması [25]

SPARQL ile sorgulama yapmak için dört ana sorgu formu bulunmaktadır:

- **SELECT:** Sorgulamalarda en fazla kullanılan formdur. Bir veya birden fazla veri setinde bulunan verilerin, belirlenen değişkenlere göre sorgulanmasına olanak sağlar. Çizelge 3.9’da bir RDF Turtle formatında örnek veri görülmektedir.

Çizelge 3.9 Örnek RDF turtle verisi [25]

```
@prefix ab: <http://learningsparql.com/ns/addressbook#> .
```

```
ab:richard ab:homeTel "(229) 276-5135" .  
ab:richard ab:email "richard49@hotmail.com" .  
ab:cindy ab:homeTel "(245) 646-5488" .  
ab:cindy ab:email "cindym@gmail.com" .  
ab:craig ab:homeTel "(194) 966-1505" .  
ab:craig ab:email "craigellis@yahoo.com" .  
ab:craig ab:email "c.ellis@usairwaysgroup.com" .
```

Bu veriye,

```
SELECT ?craigEmail  
WHERE { <http://learningsparql.com/ns/addressbook#craig>  
<http://learningsparql.com/ns/addressbook#email>  
?craigEmail . }
```

SPARQL sorgusu uygulanırsa cevap olarak,

```
-----  
| craigEmail |  
=====
```

"c.ellis@usairwaysgroup.com"
"craigellis@yahoo.com"

```
-----
```

yanıtı alınır. Bu sorguda adı Craig olan ve e-posta yüklemi ile ilişkili olan nesneler sorgulanmaktadır. Kısacası Craig adlı kişinin Çizelge 3.9’deki veride var olan e-posta adresleri sorulmaktadır. SELECT formu ile bu veri doğrudan elde edilmiştir [25].

- **ASK:** Veri seti üzerinde yapılacak sorguya yanıt verilip verilemeyeceğini ‘true’ veya ‘false’ belirtir. Genellikle iki farklı veri setinde aynı bilginin olup olmadığını denetlemek için kullanılır.

- **CONSTRUCT:** Sorguya yeni bir RDF grafiği olarak cevap verir. Veriyi bir şemadan diğerine çevirmek için etkin bir yoldur.
- **DESCRIBE:** Veri yayımcısının, veri seti hakkında sunduğu bilgiler cevap olarak alınır. Semantic Site Haritası ve Vold meta verileri genellikle cevap olarak geri döner.

SPARQL 1.1 versiyonunda bu dört ana sorgu formu dışında birçok farklı form da bulunmaktadır. Bunlardan 'ORDER BY' ile sorgu cevapları belirli bir sıraya konulabilir. Sorgulamalara verilen cevaplar 'LIMIT' formu ile kısıtlanabilir. 'MAX' ve 'MIN' gibi kısıtlayıcılarla da sorguya verilen yanıtlar sınırlandırılabilir. 'SERVICE' sorgu formu ile SPARQL 1.1 özelliklerini sağlayan diğer SPARQL Endpoint servislerine erişilir. 'DISTINCT' ile sorgulanan verinin sadece bir kez gösterilmesi, tekrarların silinmesi sağlanır. 'UNION' sorgu formu ise farklı veri setlerinin birleştirilmesine sağlar. Ayrıca, SPARQL veri setleri üzerinde matematiksel/fonksiyonel işlemler içeren sorgulamalar da yapılabilir [24], [25].

Birçok farklı uygulama, verilerini, arayüz sorgulama imkanı sağlayan SPARQL Endpoint olarak sunmuştur. Bunlardan en çok kullanılanı Vikipedi'nin oluşturduğu DBpedia'yi sorgulamaya yarayan arayüzlerdir⁵⁰⁵¹. DBpedia'ya benzer şekilde OS de verilerini SPARQL Endpoint olarak sunmuştur⁵². Bu arayüzler aracılığıyla, sunulan veriler sorgulanabilmekte ve uygulamalarda kullanılabilir. Veri tabanlarında SQL tarafından sunulan, sorgulanan verilerin uygulamalarda kullanılması hizmeti, Semantik Web'de SPARQL tarafından sunulmaktadır. Örneğin, DBpedia'da yapılan

```
SELECT ?elvisbday WHERE {
<http://dbpedia.org/resource/Elvis_Presley>
<http://dbpedia.org/property/dateOfBirth> ?elvisbday .}
```

sorgu ile Elvis Presley'in doğum tarihi sorgulanmaktadır. Burada Dbpedia tarafından <http://dbpedia.org/snorql/?query=SELECT+%3Felvisbday+WHERE+%7B%0D%0A%3Cht>

⁵⁰ <http://dbpedia.org/sparql>

⁵¹ <http://dbpedia.org/snorql/>

⁵² <http://data.ordnancesurvey.co.uk/datasets/os-linked-data/explorer/sparql>

[tp%3A%2F%2Fdbpedia.org%2Fresource%2FElvis_Presley%3E%0D%0A%3Chttp%3A%2F%2Fdbpedia.org%2Fproperty%2FdateOfBirth%3E+%3Felvisbday+.%0D%0A%7D](http%3A%2F%2Fdbpedia.org%2Fresource%2FElvis_Presley%3E%0D%0A%3Chttp%3A%2F%2Fdbpedia.org%2Fproperty%2FdateOfBirth%3E+%3Felvisbday+.%0D%0A%7D) verilen URI, direkt olarak uygulamalarda Elvis Presley'in doğum tarihinin kullanılmasına olanak tanır [25].

SPARQL konumsal sorgulamalara da cevap verir.

```
SELECT ?x
WHERE {
    ?x a gas:GasStation;
    georss:where ?y.
    ?y rcc:part [
        a gml:Buffer;
        gml:radius "1";
        gml:bufferGeometry [
            a gml:Point;
            gml:pos "38 -77"
        ].
    ].
}
```

sorgusu uygun veri setine uygulandığında 38° Kuzey enlemi ve 77° Batı boylamına bir mil mesafedeki benzin istasyonları sorgulanabilir [104]. SPARQL sorgularını yapısal olarak denetleyen online araçlarla, sorgu ifadesinin doğruluğu denetlenebilir⁵³⁵⁴. Bir sonraki bölümde konumsal verileri sunmak ve SPARQL ile sorgulamalarına olanak tanımak için geliştirilen GeoSPARQL standardı incelenecektir.

3.5.2 GeoSPARQL

GeoSPARQL, SPARQL protokollerini ve fonksiyonlarını tamamen kabul eden ve bu özelliklere coğrafi detay (simple feature) ekleyerek SPARQL'e konumsal boyut katan bir standarttır. OGC GeoSPARQL standardı, RDF veri modelinde coğrafi detayları, geometrileri ve konumsal ilişkileri sunabilmek için SPARQL'e sınıf, ilişki ve fonksiyon tanımlamaları eklemiştir. GeoSPARQL'in üç ana bileşeni bulunmaktadır. Bunlardan ilki

⁵³ <http://www.w3.org/2005/01/yacker/uploads/SPARQL>

⁵⁴ <http://rdf.csem.flinders.edu.au/joseki/query-validator.html>

coğrafi detayları sunmak için bir sözlük, ikincisi konumsal filtreleme fonksiyonları, sonuncu ise sorgu dönüşüm kurallarıdır [26].

GeoSPARQL'in konumları sunmak için hazırladığı RDF sözlüğü, 'Konumsal Nesne', 'Coğrafi Detay' ve 'Geometri' sınıflarından oluşmaktadır. Tüm coğrafi detaylar ve geometriler 'Konumsal Nesne' sınıfının alt sınıfı olarak hazırlanmıştır. 'Coğrafi Detay' tanımı olarak ISO 19156:2011, 'Geometri' tanımlamaları olarak da ISO 19107 standardının tanımlamaları kullanılmıştır. GeoSPARQL RDF sözlüğü kullanılarak bir coğrafi detayın geometrisi tanımlanmak istendiğinde, sözlükte tanımlanan 'geometrisiVardır' (ogc:hasGeometry) ilişkisi ile geometrinin iki boyutlu olarak GML veya WKT (ogc:asGML, ogc:asWKT) veri tipleri ile ifade edileceği belirtilmiştir. KML, GeoJSON gibi çok kullanılan formatların sözlüğe daha sonraki yıllarda ekleneceği ifade edilmiştir.

GeoSPARQL'in ikinci bileşeni konumsal filtreleme fonksiyonlarıdır. Bu fonksiyonların kullanılabilmesi için SPARQL FILTER fonksiyonu, GML literalleri (ogc:GMLLiteral) ve WKT literalleri (ogc:WKTLiteral) desteklenmelidir. Bu destekler sağlandığı takdirde, uzaklığı belirten fonksiyon 'ogcf:distance', bir çember oluşturan fonksiyon ogcf:buffer, kesişimleri sunan fonksiyon ogcf:intersection, birleşimleri sunan fonksiyon ogcf:union ve daha birçok fonksiyon kullanılabilir duruma gelmektedir.

GeoSPARQL'in son bileşeni ise sorgu dönüşüm kurallarıdır. Bu kuralların işlerlik kazanabilmesi için örneklemelerin 'ogc:SpatialObject' sınıfı ile tanımlanması ve 'ogc:asWKT' ile 'ogc:asGML' veri ilişkilerinin örneklemelerin geometri temsiliinde kullanılması gerekli koşuldur. Bu şartlar sağlandığında OGC tarafından belirlenen dokuz konumsal ilişki (eşittir, ayrıktır, kesişir, dokunur, keser, içindedir, içerir ve çakışır) işler duruma gelmektedir.

Ek F'de SPARQL ve GeoSPARQL ile yapılan örnek sorgulamalar yer almaktadır.

3.6 RDF Sözlükleri ve Veri Modelleri Arasında Bağlantılar Kurulması

Bağlı Veri yaklaşımının dördüncü ilkesi, RDF bağlantılarının Web'deki diğer veri kaynakları ile ilişkilendirilmesini öngörür. RDF bağlantıları global veri kümelerini veri uzayında birbirine bağlayan yapıştırıcıdır. Teknik olarak, RDF bağlantısı bir RDF

üçlüsüdür. Üçlünün öznesi, bir veri setinin ad alanındaki bir URI referansdır. Üçlünün yüklemi ve nesnesi diğer veri setlerinin içerisindeki URI referanslarıdır. Bu URI'leri yeniden referanslandırmak, kaynağın tanımlamalarına erişimi sağlar.

Üç tür RDF bağlantısı vardır. İlki ilişki bağlantılarıdır. Bu bağlantılar kaynaklar arasındaki ilişkileri gösterir. İkincisi kimlik (identity) bağlantılarıdır. Kimlik bağlantıları, farklı kaynaklar tarafından farklı URI'lerle aynı varlığın URI'leri arasındaki eşleşmeyi sağlar. Örneğin, OWL'in owl:sameAs özelliği ile aynı varlığa ilişkin iki farklı URI birbirine eşitlenebilir. Son RDF bağlantısı ise RDF sözlüğü veya ontoloji ile kurulan bağlantılarıdır. Bu, kaynağın, sözlük teriminin tanımına olan bağlantısını gösterir. Sözlük bağlantıları veriyi kendini tanımlayabilir yapar ve Bağlı Veri uygulamalarının veriyi anlamasına, sözlükler arasındaki bağlantıları kurmasına olanak sağlar [14], [108].

RDF bağlantıları manuel veya otomatik yöntemlerle kurulabilir. Genellikle küçük ve statik veri setleri için manuel yöntem tercih edilirken; büyük veri setleri için otomatik yöntem uygulanır. Otomatik yaklaşımda veri setleri arasında RDF bağlantıları oluşturmak için otomatik ya da yarı otomatik kayıt bağlama (record linkage) yaklaşımından yararlanılır. İki adet kayıt bağlantı tekniği vardır. Bunlardan ilki olan anahtar kelime tabanlı (key-based) bağlama tekniğinde, iki veri kaynağının da kullandığı ortak anahtar kelimeler temel alınarak bağlantılar kurulur. İkincisi, benzerlik temelli (similarity-based) bağlantı tekniğidir. Bu yaklaşımda veri setlerindeki varlıklar karşılaştırılarak eşik değerin üzerinde benzerliğe sahip varlıklar bağlanır. Örneğin, aynı coğrafi varlık ile ilgili veri setlerinin bulunduğu DBPedia ve Geonames'de veriler; enlem, boylam, yer adı gibi özellikler açısından benzerlik karşılaştırması sonucu birbirine bağlanmaktadır. Bu yöntemi kullanan pek çok araç bulunmaktadır. Bunlardan ilki Semantik Web Bağlantı Keşfetme Çatısı⁵⁵ (A Link Discovery Framework for the Web of Data – SILK) yazılımıdır. SILK, kaynakların sahip olduğu farklı özellikleri eşleştirerek, bir benzerlik skoru oluşturur ve bu skora göre eşleşme sağlar. Google Refine yarı otomatik eşleşme yapmaya olanak tanır. SPARQL kullanarak da veri setleri arasında bağlantılar

⁵⁵ <http://wifo5-03.informatik.uni-mannheim.de/bizer/silk/>

kurulabilmektedir [109]. Bu alanda yapılan diğer çalışmalar W3C tarafından listelenmiştir⁵⁶.

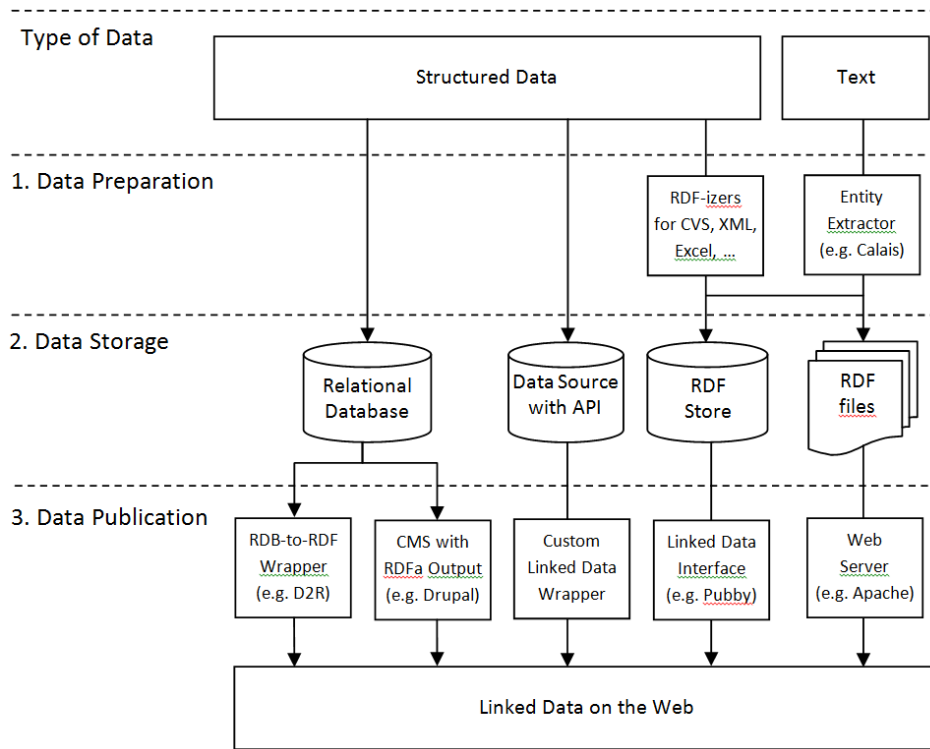
3.7 Bağlı Veri Yayımlama Yöntemleri

Önceki bölümlerde açıklandığı gibi, Semantik Web, verilerin RDF formatında yapılandırılmasını öngörür. Bu nedenle farklı formatlarda ifade edilen verilerin, ilk aşamada RDF veri modeline dönüştürülmesi gerekir. Dört farklı formattan RDF veri modeline dönüşüm yapılabilir: (i) Metinsel veriler, (ii) yapılandırılmış veriler (CSV, spreadsheet), (iii) ilişkisel veri tabanları ve (iv) API verileri.

- 1) Metinsel veriler: Open Calais ve Onto Miner gibi araçlar ile düz metin formatındaki veriler, RDF formatına dönüştürülür. % 80 başarı sağlanabilmekte, ancak manuel müdahale gerektirmektedir.
- 2) Yapılandırılmış veriler (CSV, XML, Excel): Veriler, virgülle ayrılmış bir dosya formatında ise Pearl gibi bir script dili ile RDF/XML formatına çevirilebilir. Veriler XML formatındaysa XSLT ile dönüşüm yapılır. Excel, CSV gibi formatların RDF/XML'e dönüştürülmesine olanak veren pek çok açık kaynak ve ticari yazılım vardır (örn., Cambridge Semantics, TopBraid Composer, XLWrap, Google Refine).
- 3) Veri tabanları: Veri tabanlarında tutulan veriler önce CSV ve Excel olarak formatında, ardından yukarıda belirtilen araçlarla RDF formatına dönüştürülebilir. Ancak, verilerini sürekli veri tabanında organize edenler, veri tabanının dinamik biçimde RDF'ye dönüştürülmesini isterler. Berners-Lee (1998b) ilişkisel veri tabanlarının RDF'ye dönüşümü için basit bir yaklaşım sunmuştur [110]. Bu yaklaşımda, her ilişkisel veri tabanı tablo girdisi bir RDF düğümü, her sütun bir RDF yüklemi ve her hücre bir RDF değeri olarak ifade edilebilir. Her tablo bir RDFS sınıfını ve anahtarlar da sınıflar arası ilişkileri belirler [17].

⁵⁶ <http://www.w3.org/wiki/TaskForces/CommunityProjects/LinkingOpenData/EquivalenceMining>

Bağlı Veri yayımlamak için ikinci aşama verilerin nasıl saklanacağına karar verilmesidir. İlk aşamada hazırlanan veriler, farklı şekillerde saklanabilir. Şekil 3.21’de görüldüğü gibi veriler RDF olarak, RDF saklama araçlarında (triple store), API’lerde veya ilişkisel veri tabanlarında hazır hale getirilebilmektedir. Hazırlanan veriler üç farklı yapıda Web’de yayımlanabilir. Statik yapıda, HTML yapısı kullanılarak ve ilişkisel veri tabanları kullanarak Bağlı Veri yayımlanabilir [14], [17], [111]. Bu üç farklı RDF yayım mekanizmasının dışında “RDF dump” (tüm RDF dokümanları içeren zip dosyası) veya SPARQL Endpoint (sorgulamaya olanak sağlayan bir arayüz) olarak veri setleri yayımlanabilir. Fakat sadece bu iki yol kullanılarak veri yayımlamak, Bağlı Veri yayımlama olarak düşünülmez [17].



Şekil 3.21 Bağlı Veri yayımlama yöntemleri [14]

Statik veri yayımlama, küçük miktarda veriyle uğraşırken en doğru yöntemdir. RDF dokümanları Web sunucusu üzerinden yayımlanır. MIME tipi olarak (application/rdf+xml) olarak belirlenir ve Bağlı Veri uygulaması veriyi RDF olarak algılar. Ayrıca, RDF’lerin HTML olarak da keşfine olanak sağlamak için HTML link özelliğine “<link rel="alternate" type="application/rdf+xml" href="idaribirim.rdf">” şeklinde tanımlamalar yapılmalıdır.

Bağlı Veri, HTML içine gömülü RDFa olarak yayımlanabilir. Bu yöntemin avantajı, veri değiştiği zaman sadece bir doküman ile uğraşılmasıdır. Şu anki içerik yönetim sistemlerine uygun bir yapıdır. HTML olarak üretilmiş yapılar RDFa'ya genişletilebilir yapıdadır. W3C tarafından üretilen RDFa Disteller and Parser⁵⁷ bu amaç için kullanılabilir. Veri tabanlarından direkt olarak HTML'e dönüşüm, RDFa için uygun değildir. Son dönemde Google, Yahoo gibi büyük arama motoru şirketleri RDFa verileri kullanarak, arama sonuçlarını iyileştirmeye çalışmakta ve RDFa ile küçük bilgiler (rich snippets) sunmaktadır.

Bir diğer yöntem, ilişkisel veri tabanlarından Bağlı Veri yayımlamadır. Veri büyük, sıkça değişen, güncellemeye ihtiyaç duyan yapıdaysa ve HTML'e bağlı değilse en uygun yaklaşımın bu olduğu ifade edilmiştir [17]. Fakat veri tabanları ve RDF arasında yapısal farklılıklar olduğundan, bu dönüşümde sorunlar görülmektedir. Veri tabanlarında yer alan ilişkiler, sadece RDF dokümanları tanımlamaları ile tam olarak eşleşmemektedir. Bu soruna çözüm sunmak için W3C bir çalışma grubu kurmuştur fakat grup, dönüşüm için henüz bir standart şema oluşturamamıştır [112].

Veri tabanlarını Bağlı Veri olarak yayımlamak için birçok farklı çalışma yapılmıştır [113]. Bu çalışmaların ortak noktası, veri tabanları ve Bağlı Veri arasında bir eşleştirme (mapping) yapılmasıdır. İlişkisel veri tabanlarını Bağlı Veri olarak yayımlayabilen araçlar şunlardır:

- OpenLink Virtuoso: Veri tabanı sütunlarını RDF üçlülerine çevirmek için 'Virtuoso Meta Schema Language' kullanılarak bir 'quad map patterns' oluşturulur [85], [86].
- Triplify: Web uygulamaları için bir yazılım eklentisidir. SQL sorguları üzerindeki HTTP get isteklerini haritalar, sorgular ve RDF üçlülerini olarak yayımlar. Verileri RDF/XML, JSON ve Bağlı Veri formatlarında yayım imkanı sağlar. SPARQL sorgularını desteklemez [114].

⁵⁷ <http://www.w3.org/2012/pyRdfa/>

- R2O⁵⁸: XML tabanlıdır. İspanya'nın GeoLinkedData projesinde üretilmiştir. Doğrudan Bağlı Veri üretmez. Ürettiği statik RDF'leri, Virtuoso triple store üzerinden Web'de yayımlar.
- D2R Sunucusu: Bağlı Veriye erişim sağlayan bir tarayıcı ile (tabulator, disco) RDF tanımlamalarına Web üzerinden bir URI'yle erişilebilir. D2R kaynakların RDF tanımlamalarına HTTP üzerinden erişime olanak sağlar. Ayrıca, HTML arayüzü ve SPARQL Endpoint sağlamaktadır. D2R kendi haritalama dili olan D2RQ dilini kullanır. RDFS ile veri tabanı arasında ilişkiler tanımlanır.
- R2RML (Cyganiak 2011): Veri tabanından RDF'ye standart bir eşleştirme (mapping) sağlamak için W3C tarafından yürütülen çalışmadır. Henüz standartlaşmış bir çözüm sunmamıştır [112].

Son olarak, API'lerde saklanan veriler de Bağlı Veri olarak yayımlanabilmektedir. Twitter, Flickr, Facebook API'leri için bir dönüşüm scripti veya 'wrapper' yazarak, Bağlı Veri yayımlanabilmektedir. Bu dönüşüm araçları oluşturulurken, API'nin sorgulamalara vereceği yanıtlar göz önüne alınmalıdır. İlk aşamada API veriyi JSON veya XML gibi formatlara dönüştürür. Daha sonra 'wrapper' bu verileri RDF/XML formatına çevirir ve Bağlı Veri olarak yayımlar [14], [17].

3.8 Semantik Web Kapsamındaki Bağlı Veri Uygulamaları

Bu bölümde Semantik Web kapsamında, özellikle konumsal sektörde geliştirilen, uygulamalar incelenecektir. Geonames bunlardan ilkidir. Geonames, yaklaşık 8,3 milyon yer adını (toponyms) çeşitli dillerle sunan uluslararası bir projedir. Çalışmada yer adları; yer adları detay tipi (sınıf), yükseklik, nüfus ve WGS 84 datumundaki enlem, boylam koordinatlarıyla beraber sunulmaktadır. Ontolojide yönetsel birimler beş farklı seviyede ifade edilmiş ve nüfus, yükseklik, alternatif ad, Vikipedi gibi özniteliklerle tamamlanmıştır [11]. Tüm yer adlarını nokta geometrisi ile ifade eden dizin, XML ve JSON formatlarındaki API yardımıyla ve REST servisi aracılığıyla verilerini harita (Google Maps) üzerinde göstermektedir. Geonames, konumsal semantik bilgisini Web'e

⁵⁸ <http://geo.linkeddata.es/>

aktarmak için, dizindeki her yer adı için eşsiz bir URI tanımlamış ve RDF olarak sunuma hazırlamıştır. Konseptleri ve dokümanları birbirinden ayırmak için 303 see other yönlendirmesini kullanmaktadır. RDF ile tanımlanan tüm yer adları, 'Geonames Ontolojisi' ve diğer referans ontolojiler ile birlikte modellenerek Bağlı Veri olarak yayımlanmaktadır. Geonames, Semantik Web'i birbirine bağlamak için kullanılan bilgi tabanlarından DBPedia'dan sonra en çok kullanılanıdır [59].

Semantik Web kapsamında geliştirilen bir diğer uygulama OS tarafından geliştirilen yer adları dizinleridir. Birleşik Krallık Konum Programı kapsamında İdari Coğrafya ve Seçim Bölgeleri Ontolojisi, Konumsal İlişkiler Ontolojisi, Geometri Ontolojisi, Coğrafi Yer Adları Dizini Ontolojisi ve Posta Kodu Ontolojisi⁵⁹ formel biçimde tanımlanmıştır. OS, oluşturduğu ontolojileri ve başka ad alanlarında tanımlanan ontolojileri ve RDF sözlüklerini kullanarak yer adları dizinleri örneklemelerini oluşturmuştur. OS, idari sınırların ve seçim bölgelerinin geometrik ve konumsal ilişkilerinin tanımlandığı, 1/50000 ölçeğinde bir yer adları dizini tasarlamıştır. OS, yer adları dizinlerinin klasik arama yöntemleri ile kullanılmasına olanak tanıdığı gibi, Semantik Web sorgulama dili olan SPARQL ile sorgulanmasına da izin vermektedir. Konumsal veriler, GML formatında RDF dokümanlarının içinde modellenmiştir ve OS sunucuları üzerinden API'lerle yayımlanmaktadır. Birleşik Krallık sınır bilgilerinin Bağlı Veri olarak yayımlandığı yer adları dizini, Semantik Web ve Bağlı Veri ilkerine tamamen uygun tasarlanan ilk dizin olarak nitelendirilebilir. Dizinin yapılış amacı, diğer veri setleri ile bağlantılar kurarak verilerin yeni kullanım alanı elde etmesi ve OS verilerini kullanan kullanıcı sayısında artış sağlanması olarak açıklamıştır [12].

Diğer bir konumsal Semantik Web uygulaması ise Açık Sokak Haritaları (OpenStreetMap – OSM) Projesi kapsamında toplanılan verileri, Bağlı Veri Prensiplerine uygun olarak bir RDF bilgi tabanı oluşturma çabasında olan LinkedGeoData⁶⁰ girişimidir. Girişim, OSM tanımlamalarını bir ontoloji ile modellemiştir ve verileri içerik uzlaştırma ile Bağlı Veri olarak yayımlamaktadır. Veri tabanındaki verileri Bağlı Veri olarak yayımlarken Triplify yazılımı kullanılmıştır. OSM verileri, DBPedia ve Geonames ile SILK

⁵⁹ <http://data.ordnancesurvey.co.uk/ontology>

⁶⁰ <http://linkedgeodata.org/>

Bir diğ er Semantik Web uygulaması, Baėlı Veri arama motorlarıdır. Semantik Web Arama Motoru⁶⁵ (Semantic Web Search Engine – SWSE) bunlardan biridir. SWSE Google ve Yahoo gibi anahtar kelimelerle aramalarını yapar ve sonuçları bir liste olarak sunar. VisiNav⁶⁶ arama motoru ise kompleks sorgular yapma imkanı saėlayan bir arama motorudur. Semantik Web’de uygulama tabanlı arama motorları da tasarlanmıřtır. Bunlardan biri Sindice⁶⁷’dir. Baėlı Veri  zerine kurulacak uygulamaların ihtiya larını gidermek, Web’i endeksleyerek veri toplamak amacıyla geliřtirilmiřtir. Bunların yanında <http://sameas.org/> birbirine owl:sameAs iliřkisi ile baėlanmış URI’leri bulmaya yarayan bir Semantik Web uygulaması olarak hazırlanmıřtır. Britanya Yayın Kuruluřu (British Broadcasting Corporation – BBC) tarafından Semantik Web uygulaması olarak tasarlanan Web sayfaları ve ontolojiler⁶⁸, devlet verilerini A ık Baėlı Veri olarak sunan <http://data.gov.uk/> platformu, kitap ve makale gibi kaynak listelerini oluřturup y netmek i in Baėlı Veri olarak oluřturulan Talis⁶⁹ platformu ile DBPedia verilerini mobil ortamda sunan DBPedia Mobil⁷⁰ uygulaması gibi, bir ok farklı Semantik Web uygulaması bulunmaktadır.

3.9 Coėrafi Verinin Baėlı Veri Olarak Modellenmesi

Coėrafi verilerin Baėlı Veri olarak yayınlanması i in geliřtirilen *de facto* bir metodoloji olmamakla birlikte, bu alanda yapılan  alıřmalar incelendiėinde, Baėlı Veri tasarımı a benzer yolların izlendiėi g r lm řt r [12], [14], [15], [17]. Bu  alıřmalar incelendiėinde coėrafi verilerin RDF ile modellenmesi ve Baėlı Veri olarak yayımlanmasında ařaėıdaki adımların izlenilebileceėi s ylenebilir:

- 1) URI belirleme ilkelerine uygun olarak URI’ler belirlenir. foaf:name, rdfs:label gibi  zellikler kullanılarak URI’lerin adlarının ifade edilmesi, URI’lere etiket atamasının yapılması gerekmektedir. Yine, URI adlandırmalarında adları

⁶⁵ <http://www.swse.org/>

⁶⁶ <http://sw.deri.org/2009/01/visinav/>

⁶⁷ <http://sindice.com/>

⁶⁸ <http://www.bbc.co.uk/ontologies>

⁶⁹ <http://www.w3.org/2001/sw/sweo/public/UseCases/Talis/>

⁷⁰ <http://wiki.dbpedia.org/DBpediaMobile>

ontolojilerle çeşitlendirmek faydalı olabilir. Eski adı, alternatif ad, konuşma dilindeki ad, kısa ad, belirli zaman aralığındaki adlar gibi ek niteleyicilerle adlar belirtilmelidir.

- 2) Coğrafi verinin geometrisinin belirlenmesi gerekmektedir. Bunun için en yaygın olarak WGS84 RDF sözlüğü kullanılmaktadır. Fakat bu sözlükte coğrafi detayları yeterince ayrıntılı ifade edebilecek özellik bulunmamaktadır. Örneğin, bir coğrafi detay aynı anda iki nokta ile ifade edilememektedir. Bu yüzden OGC, coğrafi verilerin RDF veri modelinde ifade edilmesine olanak sağlayan GeoSPARQL sözlüğünü yayımlamıştır. GeoSPARQL’de konum bilgileri WKT ve GML formatlarında yayımlanabilmekte; geometriler nokta, çizgi, poligon olarak ifade edilebilmektedir.
- 3) Coğrafi verinin sınıflandırılması gerekmektedir. `rdf:type`, sınıflandırma için kullanılabilecek bir özelliktir. RDF sözlüklerinde yer alan sınıflara göre coğrafi veriler sınıflandırılır. Bir coğrafi detay birden fazla sınıfa sahip olabilir.
- 4) Topoloji ve mereoloji belirlenmelidir. Konum içeren Bağlı Veride topoloji ve mereoloji oldukça önemlidir. Topoloji, coğrafi detaylar arasındaki konumsal özellikleri ifade eder. Topolojik kuralları ifade eden üç farklı formel sistemden bahsedilebilir. Bunlar Boyut Olarak Genişletilmiş Dokuz Kesişim Modeli (Dimensionally Extended nine-Intersection Model - DE-9IM), Egenhofer Topolojik İlişkileri ve Alan Bağlantı Matematiği (Region Connection Calculus - RCC8) Topolojik İlişkileridir. GeoSPARQL sözlüğü, bu üç topolojik sınıflandırmanın da tanımlamalarını içermektedir. Şekil 3.23’te farklı topolojik sınıflandırmalar arasındaki ilişkiler görülmektedir. Ağ (network) topolojisi yol, akarsu gibi varlıklar arasındaki ilişkiyi belirler. Bağlı veri, çizge yapısında olduğu için bu detay tipleri kolayca ifade edilebilir. Mereoloji ise parçaları ve bütünleri belirler. Çok farklı kavramlar arasında mereolojik ilişkiler kurulmamalıdır. Genellikle aynı alandaki coğrafi detaylar arasındaki parçasallık ve bütünsellik ilişkileri mereoloji ile belirlenir.

Simple Features	RCC8	Egenhofer
equals	equals	equal
disjoint	disconnected	disjoint
intersects	$\neg$ disconnected	$\neg$ disjoint
touches	externally connected	meet
within	non-tangential proper part + tangential proper part	inside + coveredBy
contains	non-tangential proper part inverse + tangential proper part inverse	contains + covers
overlaps	partially overlapping	overlap

Şekil 3.23 Topolojik tanımlamalar arasındaki eşitlikler [26]

Buraya kadar coğrafi veri şekillendirilmiş ve Bağlı Veri olarak yayımlanmaya hazır hale getirilmiştir. Bu aşamadan sonra Bağlı Veri yayımlamak için tasarıma başlanabilir.

- 1) Bağlı Verinin ne hakkında olduğu, amacı ve kapsamı belirlenir. Burada, Bağlı Veri yayımlamak için oluşturulacak olan ontoloji de düşünülerek, kapsam belirlenir. Oluşturulacak olan ontoloji ile hazırlanan coğrafi veriye uyup uymadığı kontrol edilir. Coğrafi verinin SPARQL sorgulamalarına uygun cevaplar verip veremeyeceği, ve tekrar kullanılabileceği alanlar düşünülür. Daha sonra coğrafi veriyi ifade edecek ontoloji taslak olarak hazırlanır. Ontolojideki kavramlar belirlenir. Ontolojinin yeniden kullanımına olanak vermek için genel tanımlamalar yapılmalı ve rdfs:domain ve rdfs:range gibi kısıtlayıcı özelliklere mümkün olduğunca yer verilmemelidir. RDFS ontolojisinin hangi sorulara cevap vereceği belirlendikten sonra ontolojide rdfs:comment özelliği kullanılarak ontolojinin amacı ve kapsamı belirtilebilir.
- 2) Coğrafi veri taslağı, hazırlanan ontolojiye uygun hale getirilir. Veri tabanı, CSV, spreadsheet, shapefile, Map Info tab yapılarındaki veriler, RDF yapısına uygun hale getirilir. RDF için uygun ontolojiler ile RDF sözlükleri seçilir ve veriler farklı ontolojilerle birlikte modellenir.
- 3) Ontolojideki sınıflar, özellikler kesin olarak belirlenir. Kullanılacak başka ontoloji tanımlamaları, yapılacak olan ontolojinin içerisine aktararak kullanılabilir. Ontoloji tanımlamaları yapılırken, ontolojinin yeniden kullanılabilmesini

sağlamak için aşırı tanımlamalar yapılmamalıdır. Ontoloji modellemede Protege, Neologism gibi araçlar kullanılabilir ya da <https://jena.apache.org/> gibi API'ler yardımıyla ontolojiler tasarlanabilir. Bunlar haricinde, ontolojinin Bağlı Veri prensiplerine uygun olmasına dikkat edilmelidir. Ontolojideki tüm sınıf, ilişki, öznitelik tanımlamaları yeniden referanslandırılabilir (dereferencable) olmalıdır. Başka sözlüklerin URI'lerine direkt olarak müdahale edilemediği için, bu URI'lerin ontolojilerde kullanılması önerilmemektedir. Doğrudan URI'yi kullanmak yerine owl:sameAs, owl:equivalentProperty, rdfs:seeAlso gibi özelliklerle diğer sözlüklerin benzer URI'leri ile eşitlik kurulabilir.

- 4) URI'lerin belirlenmesi gerekmektedir. Gerçek dünya nesneleri için bir URI, HTML sayfası için bir URI, RDF/XML sunumu için ayrı bir URI tanımlaması gerekmektedir.
- 5) URI tanımlamaları yapıldıktan sonra Bağlı Verinin hangi yöntemle yayımlanacağına karar verilip, o yöntemin gerekleri doğrultusunda coğrafi veriler Bağlı Veri olarak yayımlanabilir.

Bir sonraki bölümde Türkiye İdari Birimler Dizini için Semantik Web kapsamında ontolojiler hazırlanacaktır. Daha sonra bu ontolojilere uygun, RDF örneklemeleri hazırlanacak ve Bağlı Veri olarak yayımlanacaktır.

TÜRKİYE İDARİ BİRİMLER DİZİNİ İÇİN BİR MODEL ÖNERİSİ VE DİZİNİN İSTANBUL ÖRNEĞİNDE BAĞLI VERİ OLARAK YAYIMLANMASI

Ülkemizde <http://nisanyanmap.com/> gibi resmi olmayan, özel konulara yönelmiş olan yer adları dizinleri bulunduğu gibi, resmi kurumlar tarafından yayımlanan yer adı dizin çalışmaları da bulunmaktadır. En önemlisi HGK tarafından ‘1/250.000 Ölçekli Türkiye ve Çevresi Coğrafi Ad Dizini’ ile bu dizine idari birim detayı açısından veri sağlayan ‘Türkiye Yerleşim Yerleri Veri Tabanı’⁷¹’dir. 1/250.000 Ölçekli Türkiye ve Çevresi Coğrafi Ad Dizini; Türkiye ve çevresine ait yerleşim yerleri, topoğrafik ve hidrografik detayların adları, türleri ve coğrafi koordinatlarını basılı formatta sunmaktadır. Türkiye Yerleşim Yerleri Veri Tabanı ise yerleşim yerlerinin resmi adı, 1/25.000 ölçekli paftası ve grid numaraları, coğrafi koordinatları ve yüksekliğine ilişkin verileri içermektedir. İçişleri Bakanlığı da benzer işlevlere sahip ‘Mülki İdare Birimleri Envanteri’⁷²’ni oluşturmuştur. Bu envanter; idari birim adlarını basit bir liste biçiminde sunmakta, Web ara yüzü ile sorgulanan birimin bağlı olduğu birim adını ve türünü göstermektedir. Ayrıca yetkili kullanıcılar, sorguladıkları birimlerin konumlarını bir harita üzerinde görebilmektedirler. Kısaca açıklanan bu ulusal dizin ve envanterler, coğrafi yer adları dizinlerinin temel bileşenlerine kısmen sahip olmakla birlikte bazı yetersizlikler göstermektedir:

⁷¹ <http://www.hgk.msb.gov.tr/urunler/diger/gazetteer.html>

⁷² <https://www.e-icisleri.gov.tr/Anasayfa/MulkildariBolumleri.aspx>

- Dizinler; detayların alternatif adlarını, zamansal faktörleri ve referans kaynaklarına ilişkin öznitelikleri sunmamaktadır.
- Dizinlerde, idari birim sınırlarına ilişkin konum verisi bulunmamaktadır. Çoklu doğru veya alan geometrisi ile temsil edilmesi gereken idari birimler, nokta geometrisi ile temsil edilmektedir.
- Dizinler, detaylar arasındaki topolojik ilişkileri tanımlamamaktadır.

Günümüzde konumla ilişkili birçok farklı alanda yer adları dizini tasarlanmaktadır. Bu alanlardan biri, idari birimlerin sınırlarının ve özniteliklerinin beraber sunulduğu idari birim yer adları dizinleridir. Son dönemde, yer adları dizinleri Semantik Web kapsamında şekillendirilmekte ve Bağlı Veri olarak yayımlanmaktadır. Tez kapsamında, ülkemizdeki idari birimlere ilişkin bir yer adları dizini için ontolojik bir model geliştirilmesi, geliştirilen modelin İstanbul ölçeğinde Bağlı Veri olarak yayımlanması amaçlanmaktadır. Bu amaca ulaşabilmek için ülkemizdeki idari birimler mevzuatı incelenip, İdari Birim Ontolojisinin kavramsal altyapısını modellemek, ilk adım olarak belirlenmiştir (bkz. Bölüm 4.1). İkinci adımda, mevzuatta belirlenen kavramlardan ve farklı ontolojilerden yararlanarak hazırlanan, İdari Birim Ontolojisinin modellenme süreci anlatılacaktır (bkz. Bölüm 4.2). İzleyen bölümde, genel olarak kabul görmüş ve önceki bölümde hazırlanan ontolojiler kullanılarak, İstanbul İli, Beşiktaş İlçesi içindeki idari birimler için RDF dokümanları hazırlanacak ve Web’de sunulacaktır. RDF dokümanlarının hazırlanması için kullanılacak veriler İstanbul Büyükşehir Belediyesinden edinilen coğrafi veri tabanı üzerinde yapılacak analizlerle sağlanacak (bkz. Bölüm 4.3.1); Beşiktaş İlçesi ve idari birimleri için URI tanımlamaları yapılacaktır (bkz. Bölüm 4.3.2); elde edilen veriler ve oluşturulan URI’lerle birlikte, uygun ontolojiler kullanılarak RDF dokümanları hazırlanacaktır (bkz. Bölüm 4.3.3). Ardından hazırlanan RDF dokümanları Bağlı Veri olarak Web’de yayımlanacak (bkz. Bölüm 4.3.4) ve Bağlı Verilerin keşfini kolaylaştırmak için, Web sunumu yapılacaktır (bkz. Bölüm 4.3.5).

4.1 Mevzuat Analizi

11. yüzyılda Türkiye ismiyle anılmaya başlayan Anadolu ve çevresinin, mülki (merkezi yönetim) ve mahalli (yerel yönetim) coğrafya sınırları, geçmişten günümüze birçok defa

değişime uğramıştır. Osmanlı İmparatorluğunun kuruluşundan yıkılışına değin mülki idari birimlerinin organizasyonunda önemli değişiklikler olmuştur. Kuruluş döneminde sadece kazalardan oluşan idari birimlere, gelişme döneminde sancak ve beylerbeylikler eklenmiştir. Gelişme döneminde eyaletlerin sancaklara, sancakların kazalara, kazaların da nahiye ve köylere ayrıldığı mülki birimler, Tanzimatın ilanına dek varlıklarını sürmüştür. Tanzimat döneminde yayımlanan ‘Vilayet Nizamnamesi’ (1864) ve ‘İdare-i Umumiye-i Vilayet Nizamnamesi’ (1871) eyalet yönetimini kaldırmış; mülki birimleri vilayet, sancak (liva), kaza, nahiye ve köy biçiminde tanımlayarak; valiyi vilayetin en üst düzey yöneticisi konumuna getirmiş ve il genel meclislerini oluşturmuştur [116], [117], [118]. 1876 yılında yürürlüğe giren ‘Kanun-i Esasi’ ise mülki idari birimlerini, vilayet, livaya ve kaza olarak değiştirmiştir. Son olarak 1913 yılında yürürlüğe giren ‘İdare-i Umumiye-i Vilayet Kanunu Muvakkat’ ile vilayet encümenine bazı yetkiler verilmiş ve il merkezinde merkez ilçe kurulması öngörülmüştür. Ortaylı’ya (1976) göre; Türkiye Cumhuriyeti mülki idari birimlerinde, Tanzimat Fermanı ile tanımlanan mülki idare birimleri büyük ölçüde esas almıştır. Cumhuriyet dönemindeki mülki ve mahalli birimler aşağıda özetlenmiştir [119].

1982 Anayasası’nın 123’üncü maddesinde belirtilen *‘İdarenin kuruluş ve görevleri, merkezden yönetim ve yerinden yönetim esaslarına dayanır’* hükmü, ülkemizde merkezden ve yerinden yönetim olarak iki idare örgütlenmesi oluşturmaktadır. Başkent ve taşra olmak üzere iki bölüme ayrılan merkezi yönetim; cumhurbaşkanlığı, başbakanlık, bakanlar kurulu ve başbakanlık ile bakanlıklara bağlı bazı merkez ve taşra teşkilatlarından oluşmaktadır. Başkent örgütünü, cumhurbaşkanı, başbakan, bakanlar ve yardımcı kuruluşlar oluştururken; taşra örgütünü, iller, ilçeler ve bucaklar oluşturmaktadır.

1982 Anayasası’nın 126’ncı maddesi, merkezi idare örgütlenmesini şöyle öngörmektedir: *‘Türkiye, merkezî idare kuruluşu bakımından, coğrafya durumuna, ekonomik şartlara ve kamu hizmetlerinin gereklerine göre, illere; iller de diğer kademeli bölümlere ayrılır’*. Burada sözü edilen diğer kademeli bölümler ilçe ve beldeyi ifade etmektedir.

Merkezi yönetimin taşra örgütü olarak il, 5442 sayılı ‘İl İdaresi Kanunu’ ile yönetilmektedir. Bu kanuna göre iller, ilçelerden oluşmakta; bir ilin veya ilçenin kurulup kaldırılması İçişleri Bakanlığı kararı ve Cumhurbaşkanı onayı ile gerçekleştirilmektedir (mad. 1). ‘İl genel idaresinin başı ve mercii validir’ (mad. 4). Vali hem devleti, hem de hükümeti temsil eder; yetki alanı, il mülki sınırı içinde kalan alandır [120]. İkincil düzeydeki taşra örgütü ilçe ise, kaymakam tarafından yönetilir (mad. 27). Kaymakam, bir hükümet temsilcisidir ve valinin kontrolü altındadır; yetki alanı, ilçe mülki sınırı içinde kalan alandır.

Yerel (mahalli) yönetimler, 1982 Anayasasının 127’inci maddesi ile tanımlanmıştır: ‘Mahallî idareler; il, belediye veya köy halkının mahallî müşterek ihtiyaçlarını karşılamak üzere kuruluş esasları kanunla belirtilen ve karar organları, gene kanunda gösterilen, seçmenler tarafından seçilerek oluşturulan kamu tüzel kişileridir’. Yerel yönetimler il, belediye ve köy olarak belirlenmiştir.

İlin yerel yönetim birimi, 5302 sayılı ‘İl Özel İdaresi Kanunu’ ile oluşturulan il özel idareleridir. İl özel idaresi, ‘il halkının mahallî müşterek nitelikteki ihtiyaçlarını karşılamak üzere kurulan ve karar organı seçmenler tarafından seçilerek oluşturulan, idarî ve malî özerkliğe sahip kamu tüzel kişisi’dir (5302 sayılı kanun, mad. 3). İl özel idaresinin organları, il genel meclisi, il encümeni ve validir. ‘İl özel idaresinin görev alanı il sınırlarını kapsar’ (5302 sayılı kanun, mad. 5), il özel idarelerinin yetki alanı il mülki sınırı ile çıkarılır. Ancak, 2012 yılında yürürlüğe giren 6360 sayılı ‘On dört ilde Büyükşehir Belediyesi ve Yirmi Yedi İlçe Kurulması ile Bazı Kanun ve Kanun Hükmünde Kararnamelerde Değişiklik Yapılmasına Dair Kanun’, büyükşehirlerde il özel idaresinin tüzel kişiliğini kaldırmıştır.

İkinci düzeydeki yerel yönetim birimi olan belediye ise 1580 sayılı Belediye Kanunu ile düzenlenmiştir. Bu kanun 2005 yılına kadar yürürlükte kalmış daha sonra yerini 5393 sayılı Belediye Kanununa bırakmıştır. Belediye, ‘belde sakinlerinin mahallî müşterek nitelikteki ihtiyaçlarını karşılamak üzere kurulan ve karar organı seçmenler tarafından seçilerek oluşturulan, idarî ve malî özerkliğe sahip kamu tüzel kişisi’dir (5393 sayılı kanun, mad. 5). Belediyenin organları, belediye meclisi, belediye encümeni ve belediye başkanıdır. Belediyeler, büyükşehir belediyesi ve belediye olmak üzere ayrılır. Belde

belediyesi ise ‘belediyesi bulunan yerleşim yeri’ olarak tanımlanmıştır (5393 sayılı kanun, mad. 5). 6360 sayılı Kanun ile büyükşehir olan illerde belde belediyeleri kaldırılmıştır. Ancak, büyükşehir olmayan illerde, ilçe mülki sınırları içerisinde ilçe belediyesinin yanı sıra belde belediyesi de bulunabilir.

5393 sayılı Kanunun, belediye (ilçe belediyesi, belde belediyesi) sınırlarının çizilmesi, sınırların kesinleşmesi ve sınır uyuşmazlıklarının çözümüne ilişkin hükümler içeren beşinci, altıncı ve yedinci maddelerinin yorumlanması sonucunda; belediye sınırının mülki sınırla çakışmasının zorunlu olmadığı anlaşılabılır. Mülki idare sınırları 5442 sayılı Kanuna göre çizilmekte ve herhangi bir sınır anlaşmazlığı durumunda ise ‘Sınır Anlaşmazlığı, Mülki Ayrılma Ve Birleşme İle Köy Kurulması Ve Kaldırılması Hakkında Yönetmelik’ hükümlerine göre davranılmaktadır. Buna göre; il veya ilçe mülki sınırlarının belediye sınırları ile çakışması zorunluluğu bulunmamaktadır: İl ve ilçe sınırları ile belediye sınırları çakışabilir veya çakışmayabilir.

Belediye başkanı, belediye encümeni ve belediye meclisi organlarından oluşan büyükşehir belediyeleri, 2004 yılına dek 3030 sayılı ‘Büyükşehir Belediyesi Kanunu’ ile yönetilmekteydi. Ancak, 2004 yılında yürürlüğe giren 5216 sayılı ‘Büyükşehir Belediyesi Kanunu’ ve 2012 yılında yürürlüğe giren 6360 sayılı ‘On dört İlde Büyükşehir Belediyesi ve Yirmi Yedi İlçe Kurulması ile Bazı Kanun ve Kanun Hükmünde Kararnamelerde Değişiklik Yapılmasına Dair Kanun’ büyükşehir belediyeleri ile ilgili önemli değişiklikler yapmışlardır.

5216 sayılı Kanunun üçüncü maddesinde, ilçe belediyesi ‘büyükşehir belediyesi sınırları içinde kalan ilçe belediyesi’, ilk kademe veya belde belediyesi ise, ‘büyükşehir belediye sınırları içinde ilçe kurulmaksızın oluşturulan ve büyükşehir ilçe belediyeleriyle aynı yetki, imtiyaz ve sorumluluklara sahip belediye’ olarak tanımlanmıştır. 6360 sayılı Kanun ile yapılan değişiklikler sonucunda, 14 il belediyesi büyükşehir belediyesine çevrilmiş ve ülkemizde toplam 30 büyükşehir belediyesi oluşturulmuştur. Bu yasa, büyükşehir belediyelerinin idari sınırlarını, il mülki sınırları olarak tanımlamıştır (mad. 1). Ayrıca, büyükşehir belediyelerinin mülki sınırı içinde belde ve köy örgütleri kaldırılmış, yasa ile büyükşehire dönüştürülen illerdeki köylerin mahalle olarak, benzer şekilde belde belediyelerinin de tek bir mahalle olarak bağlı oldukları ilçe belediyesine

bağlanması öngörülmüştür. Son olarak, İstanbul ve Kocaeli il mülki sınırları içerisinde bulunan orman köylerinin de tüzel kişiliği kaldırılmış ve bu köyler bağlı bulundukları ilçe belediyesine mahalle olarak katılmıştır. Bu yasa ile büyükşehir belediyelerinin sınırları il mülki sınırlarıyla; büyükşehir ilçe belediyelerinin sınırları ise ilçe mülki sınırlarıyla çakıştırılmıştır.

1982 Anayasası en düşük düzeydeki yerel yönetim birimini köy olarak belirtmiştir. 442 sayılı ‘Köy Kanunu’ hükümlerine göre yönetilen köyler, ‘nüfusu 2000’in altındaki yerler’ olarak tanımlanmıştır (mad. 1). Köy sınırlarının nasıl çizileceği, sınır anlaşmazlıklarında neler yapılacağı Köy Kanunu ile belirlenmiştir (bkz. mad. 3-5). Bir köy sınırları içinde birden fazla mahalle bulunabilir. Ayrıca köyler arasında boşluklar da bulunabilir. Köyün yöneticisi muhtardır (mad. 10). Muhtarların bütün işlemleri kaymakam ve valinin vesayet denetimine tabidir. Yukarıda sözü edilen 6360 sayılı Kanun köyleri de etkilemiştir. Örneğin, büyükşehir belediyesi bulunan otuz ilde köylerin tüzel kişiliği kaldırılmış, bunlar bağlı bulundukları ilçenin mahallesi olmuşlardır. Ayrıca, 3152 sayılı İçişleri Bakanlığı Teşkilat ve Görevleri Hakkındaki Kanun ile coğrafi yerlerin (köy, önemli mevki ve tabii yerler) adlarının değiştirilmesine ait işlemlerde İçişleri Bakanlığına bağlı İller İdaresi Genel Müdürlüğü yetkili kılınmıştır [121].

Mahalle, ‘belediye sınırları içinde, ihtiyaç ve öncelikleri benzer özellikler gösteren ve sakinleri arasında komşuluk ilişkisi bulunan idarî birim’dir (5393 sayılı Kanun). Mahalle, muhtar ve ihtiyar heyeti tarafından yönetilir. Mahalle sınırları, ‘belediye meclisinin kararı ve kaymakamın görüşü üzerine valinin onayı ile belirlenir’ (5393 sayılı Kanun, mad. 9). Muhtar, kaymakam ve valinin denetimine tabiidir. Büyükşehir belediyelerinde mahalle en küçük yerel yönetim birimiyken; il belediyelerinde en küçük yönetim birimi mahalle ve köydür.

Sonuç olarak, yürürlükteki mevzuata göre idari birimler, mülki ve mahalli idare birimleri olarak sınıflandırılabilir. Mülki idare, il ve ilçe mülki sınırlarının kapsadığı alanlardır. İl mülki sınırı içerisinde vali, ilçe mülki sınırları içinde ise kaymakam yetkilidir. Mahalli idare birimleri ise il özel idaresi, belediye ve köy olarak belirlenmiştir. Büyükşehir dışındaki illerde, ilçe mülki sınırlarının birleşerek oluşturduğu il mülki sınırı içinde il özel idareleri yetkilidir. Belediye ise, büyükşehir belediyesi ve belediye olarak ikiye

ayrılmaktadır. Büyükşehir belediyelerinde il mülki sınırı aynı zamanda büyükşehir belediyesi sınırı olmakla birlikte; ilçe mülki sınırı da büyükşehir ilçe belediyesi ile aynı sınırı paylaşmaktadır. Bu durumda büyükşehir ilçe belediyelerini sınırları birleşince büyükşehir belediyesinin sınırlarını oluşturduğu söylenebilir. Büyükşehir dışındaki illerde ise durum farklılaşmaktadır. İlçe belediyelerinin sınırı ilçe mülki sınırı ile çakışmak zorunda değildir. İlçe mülki sınırı içinde bir ilçe belediyesi olabileceği gibi, belde belediyesi de bulunabilir. En küçük sınırlara sahip yerel yönetim birimi köy olarak belirlenmiştir. Büyükşehir belediyelerinde ilçeler sadece mahallelerden oluşmakta iken, belediyelerde mahalle ve köylerden oluşmaktadır.

Mevzuat analizi sonucu ulaşılan sonuçlar şöyle özetlenebilir:

- 1) İdari birimler, mülki ve mahalli idare birimlerinden oluşmaktadır.
- 2) Mülki idare birimleri; il, ilçe ve bucakları içerir. Ancak, bucaklar, pratikte bulunmamaktadır [122].
- 3) İlçe mülki sınırları birleşerek il mülki sınırını oluşturur.
- 4) Mahalli idare birimleri; büyükşehir olmayan illerde il özel idaresi, ilçe belediyeleri, belde belediyeleri, köyler ve mahallelerden oluşur. Büyükşehirlerde ise sadece büyükşehir belediyesi, ilçe belediyeleri ve mahalleler bulunur.
- 5) Büyükşehir olmayan illerde il mülki sınırı ile il özel idaresi sınırı; büyükşehirlerde ise il mülki idare sınırı ile büyükşehir belediyesi sınırı çakışıktır.
- 6) Her ilçede bir ilçe belediyesi bulunur. Büyükşehir belediyesi olan illerdeki ilçe mülki sınırları ile ilçe belediyelerinin sınırları çakıştır. Büyükşehir olmayan illerde ise ilçe mülki sınırları içerisinde bir ilçe ve bir veya birden çok belde belediyesi bulunabilir.
- 7) En küçük mahalli idare birimi büyükşehir olmayan illerde köy ve mahalle; büyükşehir olan illerde ise sadece mahalledir. Büyükşehirlerde köy bulunmamaktadır.

Çizelge 4.1, mevzuattan derlenen bilgilerle oluşturulan mülki ve mahalli idare birimlerinin, büyükşehir olan ve olmayan illere göre sınıflandırılmasını göstermektedir.

Çizelge 4.1 Türkiye’deki mülki ve mahalli idare birimleri

	Mülki	Mahalli
Büyükşehir Belediyesi	il ilçe	Büyükşehir Belediyesi Büyükşehir İlçe Belediyesi Mahalle
Belediye (İl Belediyesi)	il ilçe	Merkez Belediyesi İlçe Belediyesi Belde Belediyesi Köy Mahalle

4.2 Türkiye İdari Birimler Ontolojisi

Ontoloji geliştirmede birçok farklı metodolojiden yararlanılabilir. Bu çalışmada, Hart ve Dolbear (2012) tarafından önerilen yol izlenmiştir [17]. Yöntemin ilk ilkesi uyarınca, tasarlanacak ontolojinin amacı ve kapsamı; ülkemizdeki idari birimler ve aralarındaki ilişkilerin formel biçimde tanımlanması, idari birimlerin öznitelikleri, geometrileri ve idari birimler arasındaki topolojik ve mereolojik ilişkilerin açığa çıkarılması olarak belirlenmiştir. Yöntemdeki ikinci aşama, belirlenen alanla ilgili tüm kavramların ve kavramlar arasındaki ilişkilerin derlenmesidir. Önceki bölüm, ülkemizdeki idari birimlere ilişkin mevzuatı inceleyerek, idari birimlere ilişkin kavramlar ve kavramlar arasındaki ilişkileri açığa çıkarmıştır. Bu bölümde, idari birimler detayına özgü bir coğrafi yer adları dizininin Bağlı Veri yaklaşımıyla sunumuna olanak verecek Türkiye İdari Birimler Ontolojisi, idari birimlerin geometrilerini tanımlayacak bir Geometri Ontolojisi ve idari birimler arasındaki konumsal ilişkileri belirleyecek bir Konumsal İlişkiler Ontolojisi geliştirilmiştir.

Ülkemizde birden fazla idari birim yapılanmasının bulunmasından ve idari birimler arasındaki ilişkilerin RDFS ile ifade edilemeyecek karışıklıkta olmasından ötürü; ontoloji dili olarak OWL seçilmiştir. Ontolojiler, OWL 2.0 tanımlamalarını içeren, SWRL gibi eklentilere olanak sağlayan ve gelişmiş görsel olanaklara sahip ‘Protégé Ontology Editor’ (version 4.3) yazılımıyla geliştirilmiştir.

Önceki bölümde açıklanan ülkemizdeki idari örgütlenme, Türkiye İdari Birimler Ontolojisine şöyle aktarılmıştır: Ontolojideki ana sınıf 'İdari Birim', 'Mülki İdare' ve 'Mahalli İdare' olarak aynı hiyerarşik düzeydeki iki alt sınıfı içermektedir. Bunlardan ilki mülki idare birimlerini, ikincisi mahalli idare birimlerini temsil etmektedir. 'Mülki İdare' sınıfı aynı düzeydeki 'il' ve 'ilçe' alt sınıflarına ayrılmıştır. 'il' sınıfı, konumsal ve hiyerarşik ilişkileri ifade edebilmek için 'Büyükşehir il' ve 'Büyükşehir Olmayan il' olarak aynı düzeyde iki alt sınıfa ayrılmıştır. 'ilçe' sınıfı da benzer şekilde 'Büyükşehir ilçe' ve 'Büyükşehir Dışı ilçe' sınıflarına ayrılmıştır. 'Mahalli İdare' sınıfının alt sınıfları olarak 'il Özel İdaresi', 'Belediye', 'Köy' ve 'Mahalle' sınıfları belirlenmiştir. Bu sınıflardan 'Belediye'; 'Büyükşehir Belediyesi', 'il Belediyesi', 'Büyükşehir ilçe Belediyesi', 'ilçe Belediyesi' ve 'Belde Belediyesi' olarak aynı hiyerarşik düzeydeki alt sınıflara ayrılmıştır. Oluşturulan tüm sınıfların örneklemeleri (instances) birbirinden farklı olduğundan, tüm sınıflar birbirinden owl:disjointWith özelliğiyle ayrılmıştır. Bu hiyerarşik tanımlamalarla sınıflar arasında kısmi bir mereolojik / partonomik ilişki kurulmuştur. Sadece sınıf tanımlamalarıyla kısmi kurulan idari birimler arasındaki parça / bütün ilişkileri konumsal ilişkileri kullanılarak tamamlanmıştır. Bu tanımlamalar yapılırken, Türkiye Ulusal Coğrafi Bilgi Sistemi (TUCBS) projesi kapsamında üretilen idari birim veri temasından da faydalanılmıştır [123].

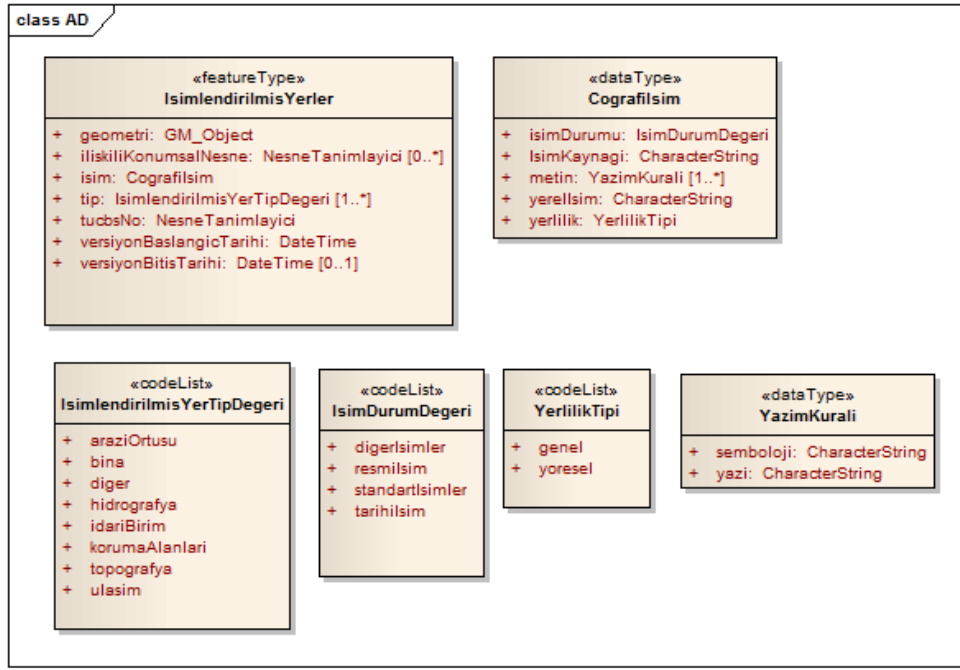
Sınıflar için yapılan ilişki ve öznitelik tanımlamalarıyla da sınıfların tüm özellikleri belirlenmiştir. Türkiye İdari Birimler Ontolojisinde, idari birimlerin veri ilişkileri, mevzuat ve literatür göz önüne alınarak şöyle belirlenmiştir:

- 1) Resmi ad,
- 2) Yerel ya da eski ad,
- 3) Geometri türü,
- 4) Konum koordinatları,
- 5) Yüzölçüm,
- 6) Nüfus,
- 7) İstatistiki Bölge Birimleri Sınıflandırması (Nomenclature of Territorial Units for Statistics – NUTS),

- 8) Posta kodu,
- 9) Ulusal Adres Veri Tabanı (UAVT) kayıtlı sabit tanıtım numarası,
- 10) İdari birimin kuruluş tarihi.

İdari birim odaklı, yer adları dizinlerini ve yer adlarını modelleyen çalışmalar ve literatür incelemesinde ortaya çıkan üç ana bileşen olan yer adı, detay sınıfı ve konumum özniteliklerinin yanına ülkemizde idari birimleri ifade eden bazı öznitelikler de ontolojiye eklenmiştir. Özniteliklerden 'NUTS' ve 'Posta kodu' bir alan belirtmesi gerekmesine rağmen, İdari Birimler Ontolojisi'nde sadece öznitelik olarak yer almaktadır. 'Posta kodu' alanları, İdari Birim Ontolojisi kapsamı dışında olduğundan başka bir ontoloji ile modellenmesi gerekmektedir. Ayrıca, ülkemizde posta kodu alanlarının yayımlandığı bir çalışma bulunmamaktadır.

Şekil 4.1'de TUCBS projesi kapsamında TUCBS Adres Veri Temasında içinde ifade edilen Coğrafi İsimler Veri Teması görülmektedir. Bölüm 2.3.5'te anlatılan INSPIRE Coğrafi Adlar Standardının ülkemize göre modellenmiş hali olan bu model sadece idari birim detayı için değil, diğer TUCBS veri modellerine kapsayacak şekilde oluşturulmuştur [124]. Burada 'IsimDurumDegeri' ve 'YerlilikTipi' sınıfları ile modellenen ad öznitelikleri, İdari Birimler Ontolojisi'nde 'Resmi Ad' ve 'Yerel Ad' ile ifade edilmeye çalışılmıştır. Adların zamansal değişimlerini TUCBS'de 'versiyonBaşlangıçTarihi' ve 'versiyonBitişTarihi' ile sağlanırken, İdari Birimler Ontolojisi'nde 'kuruluş tarihi' özniteliği ile sağlanmaya çalışılmıştır. TUCBS, coğrafi adları niteleyen konumların nokta, çizgi ve alan geometrilerine sahip olabileceğini ifade etmiştir.



Şekil 4.1 TUCBS coğrafi isim sınıfları [124]

Taslak düzeydeki Türkiye İdari Birimler Ontolojisi, bu aşamada idari birimlerin geometrilerinin ve birimler arasındaki konumsal ilişkilerin tanımlanmasına olanak veren, Geometri ve Konumsal İlişkiler Ontolojileri ile bütünlenmiştir. Oluşturulan Geometri ve Konumsal İlişkiler Ontolojileri, bu ontolojilerde tanımlanmış sınıflar, nesne ilişkileri ve veri ilişkileri Türkiye İdari Birimler Ontolojisine aktarılmıştır.

Ontoloji mühendisliğinin temel ilkesi uyarınca, Geometri ve Konumsal İlişkiler Ontolojilerinin tasarımında, öncelikle varolan ontolojilerden yararlanması gerekir. Ancak, Bölüm 3.4.4'te bir bölümü verilen bu ontolojilerin coğrafi detayları nokta geometrisi ile temsil ettiği; bunun da amaca uygun olmadığı sonucuna ulaşılmıştır. Bu nedenle, Geometri ve Konumsal İlişkiler Ontolojilerinin tasarımında, coğrafi verilerin nokta dışındaki geometrilerle temsiline olanak veren, OGC tarafından hazırlanmış GeoSPARQL RDF sözlüğü ve OS tarafından hazırlanmış Geometri Ontolojilerinden yararlanılarak, bir Geometri Ontolojisi oluşturulmuştur. Geliştirilen Geometri Ontolojisinde tüm geometrileri kapsayan 'Geometri' üst sınıfı, iki boyutlu geometrileri ifade etmek için 'koordinat' nesne ilişkisi ve koordinatların GML formatında şekillendirileceğini ifade eden 'gmlOlarak' veri ilişkisi bulunmaktadır [EK A]. Geometri ontolojisinde tanımlanan tüm sınıflar, özellikler GeoSPARQL RDF sözlüğündeki

karşılığıyla eşitlenmiştir. Fakat GeoSPARQL’de yapılan bazı tanımlamalar (WKT formatı gibi) Geometri Ontolojisinde yer almamaktadır. Bu tanımlamalar ve GeoSPARQL tarafından yapılacağı belirtilen yeni tanımlamalar, Geometri Ontolojisinin yeni versiyonlarına eklenecektir.

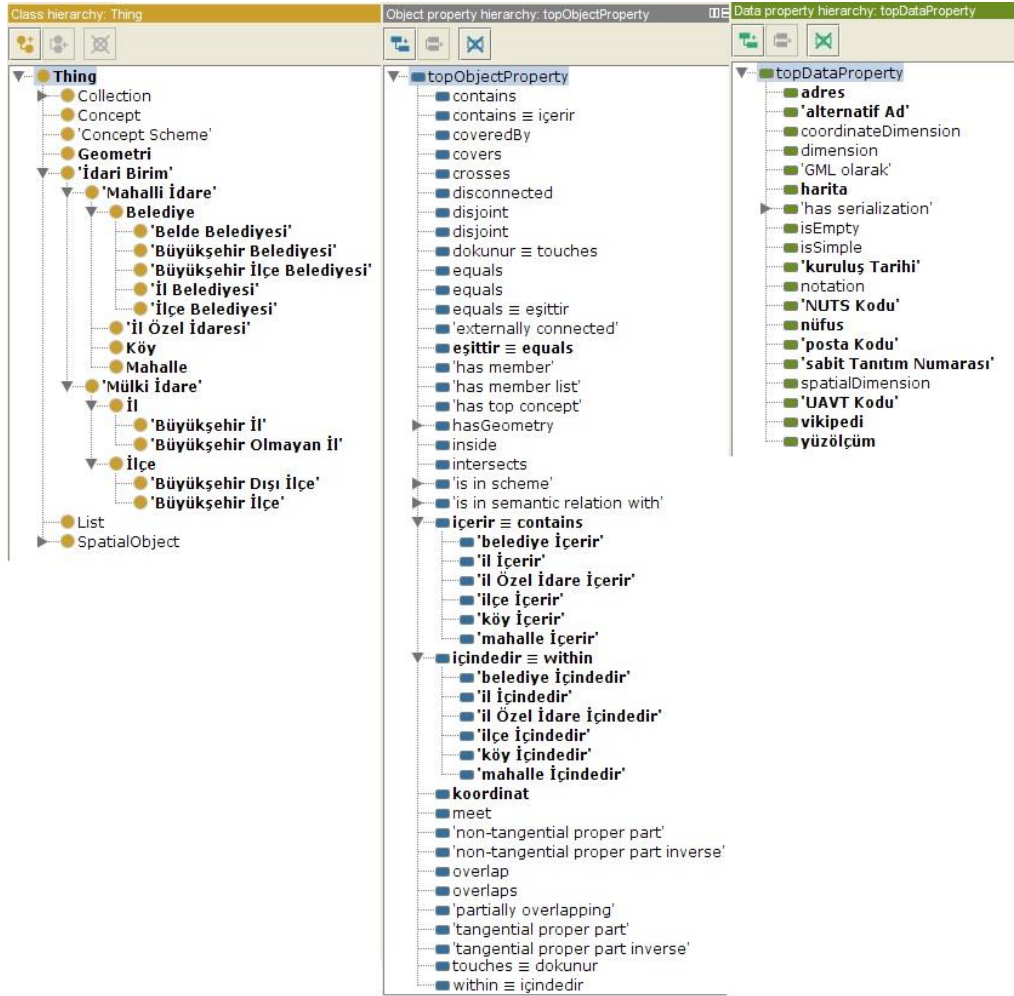
Tez çalışması kapsamında geliştirilen diğer ontoloji, Konumsal İlişkiler Ontolojisi. Konumsal İlişkiler Ontolojisi, Geometri Ontolojisine benzer şekilde GeoSPARQL tanımlamalarından yararlanılarak oluşturulmuştur. GeoSPARQL Konumsal İlişkiler Ontolojisi, OGC tarafından belirlenen 9 topolojik ilişki ile diğer standart konumsal ilişkileri tanımlamaktadır. Konumsal İlişkiler Ontolojisinde, bunlardan ‘içerir’ (contains), ‘dokunur’ (touches), ‘eşittir’ (equals) ve ‘within’ (içindedir) nesne ilişkileri tanımlanmıştır [EK B]. Bunun nedeni, Türkiye İdari Birimler Ontolojisinde kullanılacak konumsal ilişkiler için, bu dört ilişki tanımlamasının yeterli olmasıdır. OGC konumsal ilişkisinden yakın tanımlamalara sahip ‘kesişir’ (intersects) ve ‘çakışır’ (overlap) ilişkileri yerine; ‘dokunur’ konumsal ilişkisinin seçilmesinin nedeni, idari birimler arasındaki ilişkilerin ‘dokunur’ ilişki tanımlaması ile daha açık ifade edilebilmesidir. Ancak, gereksinim duyulması durumunda diğer OGC konumsal ilişkileri ve/veya RCC8, Egenhofer konumsal ilişki tanımlamaları geliştirilen ontolojiye eklenebilir. Konumsal ilişkiler ontolojisinde tanımlanan nesne ilişkileri ile GeoSPARQL tarafından tanımlanan konumsal ilişkiler arasında owl:equivalentProperty özelliği ile eşitlik sağlanmıştır. Bu bağlantıyla ontolojiler arasında eşitlik ilişkisi kurulmuştur. Bunun anlamı, Konumsal İlişkiler Ontolojisinde tanımlanan, örneğin ‘içerir’ nesne ilişkisinin, GeoSPARQL tarafından tanımlanan ‘contains’ nesne ilişkisi ile bire bir aynı olduğudur. Konumsal İlişkiler Ontolojisinde nesne ilişkilerine fonksiyonel özellikler atanmıştır. Örneğin, ‘içerir’ nesne ilişkisi geçişkendir. Yine ‘içerir’ nesne ilişkisinin ‘içindedir’ nesne ilişkisinin tersi olduğu da tanımlanmıştır. OWL 2’de tanımlanan bir özellik olan zincir aksiyomları (owl:propertyChainAxiom) da, Konumsal İlişkiler Ontolojisinde yer almıştır. Örneğin, ‘İstanbul Beşiktaş’ı içerir; Beşiktaş Beşiktaş Belediyesi ile eşittir; bu yüzden İstanbul Beşiktaş Belediyesini içerir’ konumsal çıkarımı, zincir aksiyomları ile sağlanabilmektedir.

Oluşturulan Konumsal İlişkiler ve Geometri Ontolojileri, İdari Birimler Ontolojisinde kullanılmak üzere owl:import özelliği ile İdari Birim Ontolojisine aktarılmıştır. İdari Birim Ontolojisi sınıf tanımlamlarının yapıldığı aşamada, sınıflar arasındaki mereolojik /

partonomik ilişkiler kısmi olarak olarak tanımlanmıştı. Ontolojilerde sınıflar arasındaki mereolojik / partonomik ilişkiler sadece hiyerarşik sınıf tanımlamalarıyla belirlenmemekte, sınıflarda tanımlanan nesne özellikleri tanımlamalarıyla da hiyerarşik ilişkiler kurulabilmektedir. Örneğin, 'İlçe içerir Mahalle' tanımlamasıyla, aynı düzeyde bulunan 'İlçe' ve 'Mahalle' sınıfları arasında hiyerarşik ilişki kurulmuş olur. Konumsal ilişkiler Ontolojisinde tanımlanan 'içerir' ve 'içindedir' nesne ilişkileri, İdari Birimler Ontolojisi kullanılarak hazırlanan dokümanlardan etkin sorgu sonuçları alabilmek için alt nesne ilişkilerine ayrılmıştır. Örneğin, 'içerir' nesne ilişkisi; 'il içerir', 'ilçe içerir', 'köy içerir' ve 'mahalle içerir' ilişkilerine ayrılmıştır. İdari Birimler Ontolojisindeki nesne ilişkilerin hangi sınıflar arasında kurulabileceği, rdfs:range ve rdfs:domain özellikleriyle tanımlanmıştır. Konumsal ilişkilere benzer şekilde bu nesne ilişkilerine de simetrik, geçişken gibi fonksiyonel özellik tanımlamaları yapılmıştır. Örneğin, 'ilçe içerir' nesne ilişkisi 'ilçe içindedir' nesne ilişkisinin tersi olarak tanımlanmıştır. İdari Birim Ontolojisi, ülkemizdeki iki farklı idari birim modelini de kapsayacak şekilde geliştirilmiştir. Büyükşehir olmayan illerdeki ilişkileri belirlemek için 'köy içerir', 'il Özel İdaresi içindedir' gibi nesne ilişkileri, İdari Birimler Ontolojisinde tanımlanmıştır ve ilgili sınıflara gerekli tanımlamalar yapılmıştır. Örneğin, 'Büyükşehir Olmayan İl köy içerir' ifadesi 'Büyükşehir Olmayan İl' sınıfına tanımlanmıştır.

Geometri Ontolojisinde de benzer bir yol izlenmiştir. 'koordinat' nesne ilişkisi, İdari Birimler Ontolojisinde, geometri koordinatları ifade edilmek istenen her sınıfta tanımlanmıştır. Örneğin, 'Mahalle koordinat gmlOlarak' tanımlamasıyla, 'Mahalle' sınıfının koordinatlarının 'gmlOlarak' ifade edilmesi gerektiği belirtilmiştir. Burada tanımlanan 'gmlOlarak' veri ilişkisi ise geometrinin türünü (nokta, çizgi, poligon) ve koordinatların ifade ediliş formatını belirlemektedir. İdari Birimler Ontolojisinde tanımlanan diğer veri ilişkileri de gerekli sınıflara tanımlanmıştır. Ortak olan veri ilişkileri ana sınıf için tanımlanarak, veri ilişkileri tekrardan kaçınılmıştır. Örneğin, 'Mahalli İdare' sınıfının tüm alt sınıflarında ortak olan 'kuruluş Tarihi' ve 'yüzölçümü' veri özellikler, 'Mahalli İdare' ana sınıfına atanmıştır. Bu veri ilişkileri 'Mahalli İdare' sınıfının tüm alt sınıflarına miras kalacaktır. Ancak, alt sınıflara farklı veri ilişkileri tanımlamaları da yapılmıştır. Örneğin, 'sabit Tanıtım Numarası' ve 'posta Kodu' veri özellikleri sadece 'Mahalle' sınıfına atanmıştır [EK C].

Şekil 4.2, Türkiye İdari Birimler Ontolojisi, Konumsal İlişkiler Ontolojisi ve Geometri Ontolojisi ile tanımlanan tüm sınıf, ilişki ve öznelikleri göstermektedir.



Şekil 4.2 Türkiye İdari Birimler Ontolojisi sınıf ve ilişkileri

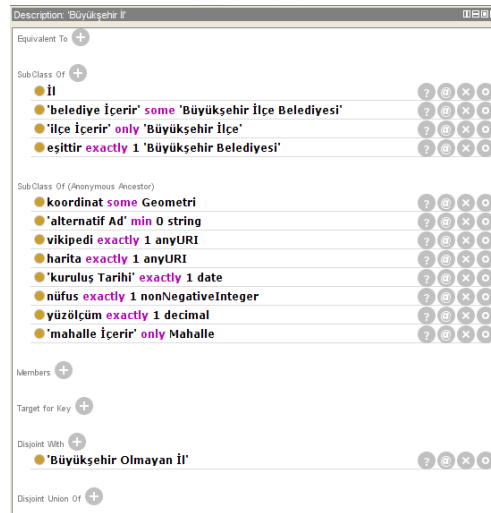
Çizelge 4.2, 'Büyükşehir İl' sınıfı için OWL formatında yapılan bazı ilişki tanımlamalarını, Şekil 4.3 ise Protégé Editöründe 'Büyükşehir İl' sınıfı için yapılan tüm tanımlamaları göstermektedir. Çizelge 4.2'de 'Büyükşehir İl' owl:Class özelliğiyle bir sınıf olarak belirlenmiş ve <http://araziyonetimi.org/ontoloji/idaribirim/BuyuksehirIl> URI'si ile tanımlanmıştır. Ardından iki ilişki tanımlaması yapılmıştır: Bu tanımlamalarda 'belediye içerir' ilişkisi ile 'Büyükşehir İl' sınıfının 'Büyükşehir İlçe Belediyesi' içerdiği; 'ilçe içerir' ilişkisi ile de 'Büyükşehir İl' sınıfının 'Büyükşehir İlçe' içerdiği belirtilmiştir. İlişki tanımlamaları bazı OWL özellikleri ile kısıtlanmıştır. İlk tanımlamada 'belediye içerir' ilişkisi owl:someValuesFrom özelliğiyle kısıtlanarak; 'belediye içerir' ilişkisinin,

'Büyükşehir İl' sınıfı için 'Büyükşehir İlçe Belediyesi' sınıfından başka sınıflarla da kurulabileceği belirtilmiştir. İkinci tanımlamada 'ilçe İçerir' ilişkisi için konulan owl:allValuesFrom özelliği ise 'ilçe İçerir' ilişkisinin, 'Büyükşehir İl' sınıfı için yalnızca 'Büyükşehir İlçe' sınıfı ile kurulabileceğini ifade etmektedir.

Çizelge 4.2 'Büyükşehir İl' sınıfı ve bazı ilişkileri

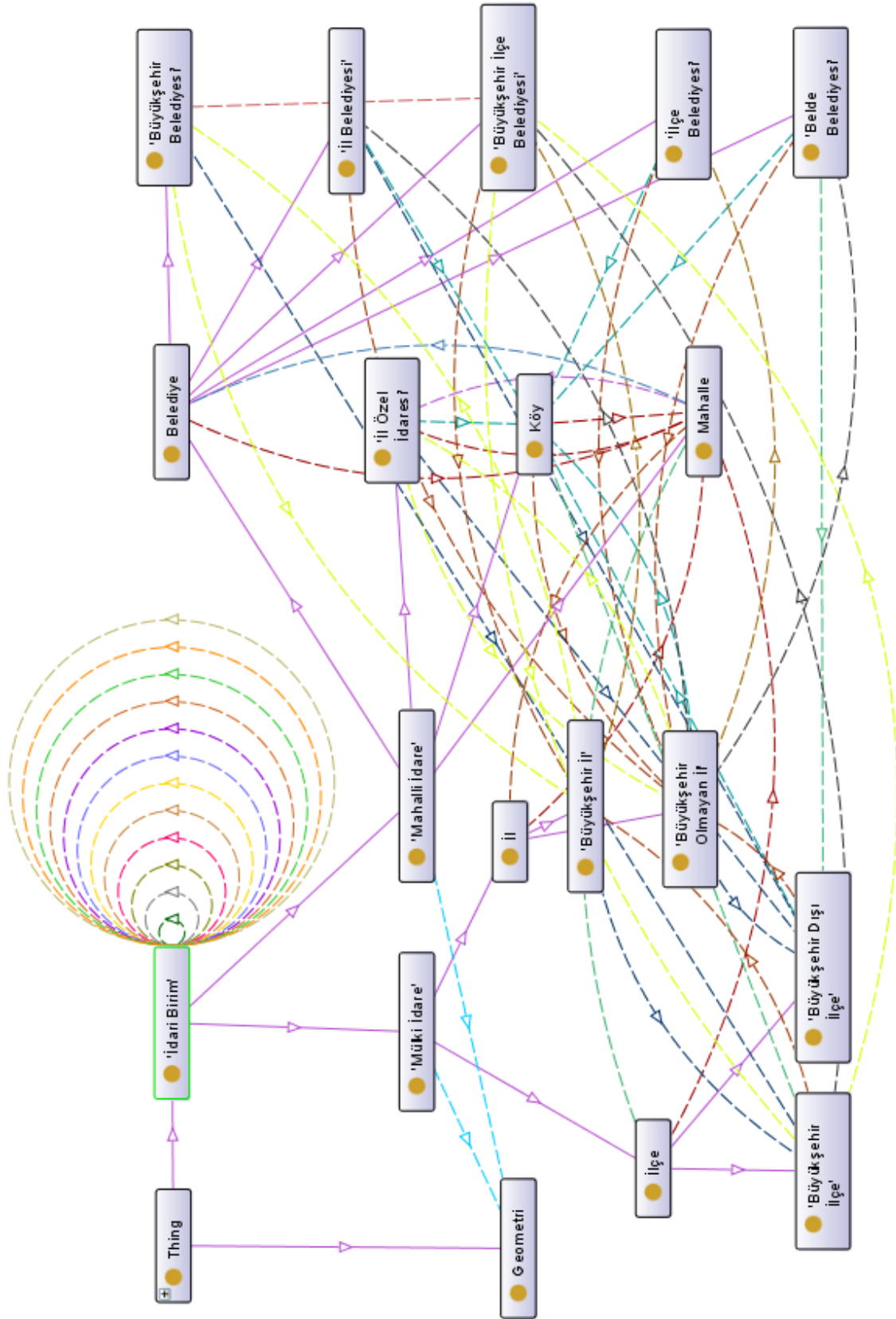
```
<owl:Class rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/BuyuksehirIl">
<rdfs:label>Büyükşehir İl</rdfs:label>
<rdfs:subClassOf rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/Il"/>
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/belediyeIcerir"/>
<owl:someValuesFrom rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/BuyuksehirIlceBelediyesi"/>
</owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf>
<owl:Restriction>
<owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/ilcelcerir"/>
<owl:allValuesFrom rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/BuyuksehirIlce"/>
</owl:Restriction>
</rdfs:subClassOf>
</owl:Class>
```

Şekil 4.4, Türkiye İdari Birimler Ontolojisi, Konumsal İlişkiler Ontolojisi ve Geometri Ontolojisindeki tüm sınıflar ve bu sınıfların birbirleriyle olan ilişkilerini göstermektedir. Şekildeki düz çizgiler sınıflar arasındaki hiyerarşiyi, kesikli çizgiler sınıflar arasında kurulmuş ilişkileri, daire şeklinde kesikli çizgilerse ilişkilerin domain ve range tanımlamalarını göstermektedir. Görselleştirme OntoGraf⁷³ ile yapılmıştır.



Şekil 4.3 Protégé editöründeki 'Büyükşehir İl' sınıfı tanımlamaları

⁷³ <https://github.com/protegeproject/ontograf>



Şekil 4.4 İdari birim, geometri ve konumsal ilişkiler ontolojileri

Tez kapsamında geliştirilmiş üç ontolojiye de <http://araziyonetimi.org/ontoloji> adresinden erişilebilir. Ontolojilerin yapısal doğrulukları, Jena Ontoloji API⁷⁴si ile test edilmiştir. Ayrıca, Jena Ontoloji API'nin sağladığı doğrulama metotlarını kullanan ve online olarak OWL'in yapısal doğruluk denetlemesini sağlayan Web siteleriyle^{75,76} de, ontolojinin yapısal doğruluğu test edilmiştir (Şekil 4.5). Ontolojilerin mantıksal doğruluğu da Fact++ gibi mantıksal çıkarsama araçları ile test edilmiş ve doğrulanmıştır.

OWL 2 Validation Report

Summary

The ontology and all of its imports are in the OWL 2 profile

Imports Closure

Ontology IRI: <http://araziyonetimi.org/ontoloji/iddaribirim/> Physical URI

Property Analysis

Non-simple properties

mahalleicerir

Explanation: 1
ilceicerir o mahalleicerir: **SubPropertyOf** mahalleicerir

mahalleicindendir

Explanation: 1
mahalleicindendir: **InverseOf** mahalleicerir
ilceicerir o mahalleicerir: **SubPropertyOf** mahalleicerir

Validation Results

Your RDF document validated successfully.

Triples of the Data Model

Number	Subject	Predicate
1	http://araziyonetimi.org/ontoloji/iddaribirim/	http://www.w3.org/1999/02/22-rdf-syntax-ns#type
2	http://araziyonetimi.org/ontoloji/iddaribirim/	http://www.w3.org/2000/01/rdf-schema#label
3	http://araziyonetimi.org/ontoloji/iddaribirim/	http://purl.org/dc/elements/1.1/creator
4	http://araziyonetimi.org/ontoloji/iddaribirim/	http://purl.org/dc/elements/1.1/contributor
5	http://araziyonetimi.org/ontoloji/iddaribirim/	http://purl.org/dc/elements/1.1/title
6	http://araziyonetimi.org/ontoloji/iddaribirim/	http://www.w3.org/2000/01/rdf-schema#comment
7	http://araziyonetimi.org/ontoloji/iddaribirim/	http://www.w3.org/2000/01/rdf-schema#isDefinedBy
8	http://araziyonetimi.org/ontoloji/iddaribirim/	http://www.w3.org/2002/07/owl#imports
9	http://araziyonetimi.org/ontoloji/iddaribirim/	http://www.w3.org/2002/07/owl#imports
10	http://purl.org/dc/elements/1.1/contributor	http://www.w3.org/1999/02/22-rdf-syntax-ns#type
11	http://purl.org/dc/elements/1.1/creator	http://www.w3.org/1999/02/22-rdf-syntax-ns#type
12	http://purl.org/dc/elements/1.1/description	http://www.w3.org/1999/02/22-rdf-syntax-ns#type
13	http://purl.org/dc/elements/1.1/title	http://www.w3.org/1999/02/22-rdf-syntax-ns#type
14	http://www.w3.org/2000/01/rdf-schema#comment	http://www.w3.org/1999/02/22-rdf-syntax-ns#type
15	http://www.w3.org/2000/01/rdf-schema#isDefinedBy	http://www.w3.org/1999/02/22-rdf-syntax-ns#type

Şekil 4.5 Ontolojilerin yapısal doğruluk testi

Tez kapsamında geliştirilen üç ontoloji ve farklı ontolojiler kullanılarak, Semantik Web kapsamında, Bağlı Veri ilkeleriyle birlikte bir pilot çalışma yapılacaktır. İzleyen bölümde, İstanbul Büyükşehir Belediyesi (İBB) Planlama Müdürlüğünden elde edilen verilerle örnek bir uygulama hazırlanacak ve Bağlı Veri olarak sunulacaktır.

4.3 Türkiye İdari Birimler Dizininin İstanbul/Beşiktaş Örneğinde Bağlı Veri Sunumu

Bu bölümde, İBB Planlama Müdürlüğünden edinilen coğrafi veri tabanı (Geodatabase – GDB) üzerinde yapılan sorgulamalar sonucunda idari birimler hakkında veri toplama, idari birimlere uygun URI atamalarının yapılması, idari birimler için RDF dokümanlarının oluşturulması ve bunların Bağlı veri olarak yayımlanması aşamaları sırasıyla anlatılacaktır.

⁷⁴ <https://jena.apache.org/documentation/ontology/>

⁷⁵ <http://mowl-power.cs.man.ac.uk:8080/validator/>

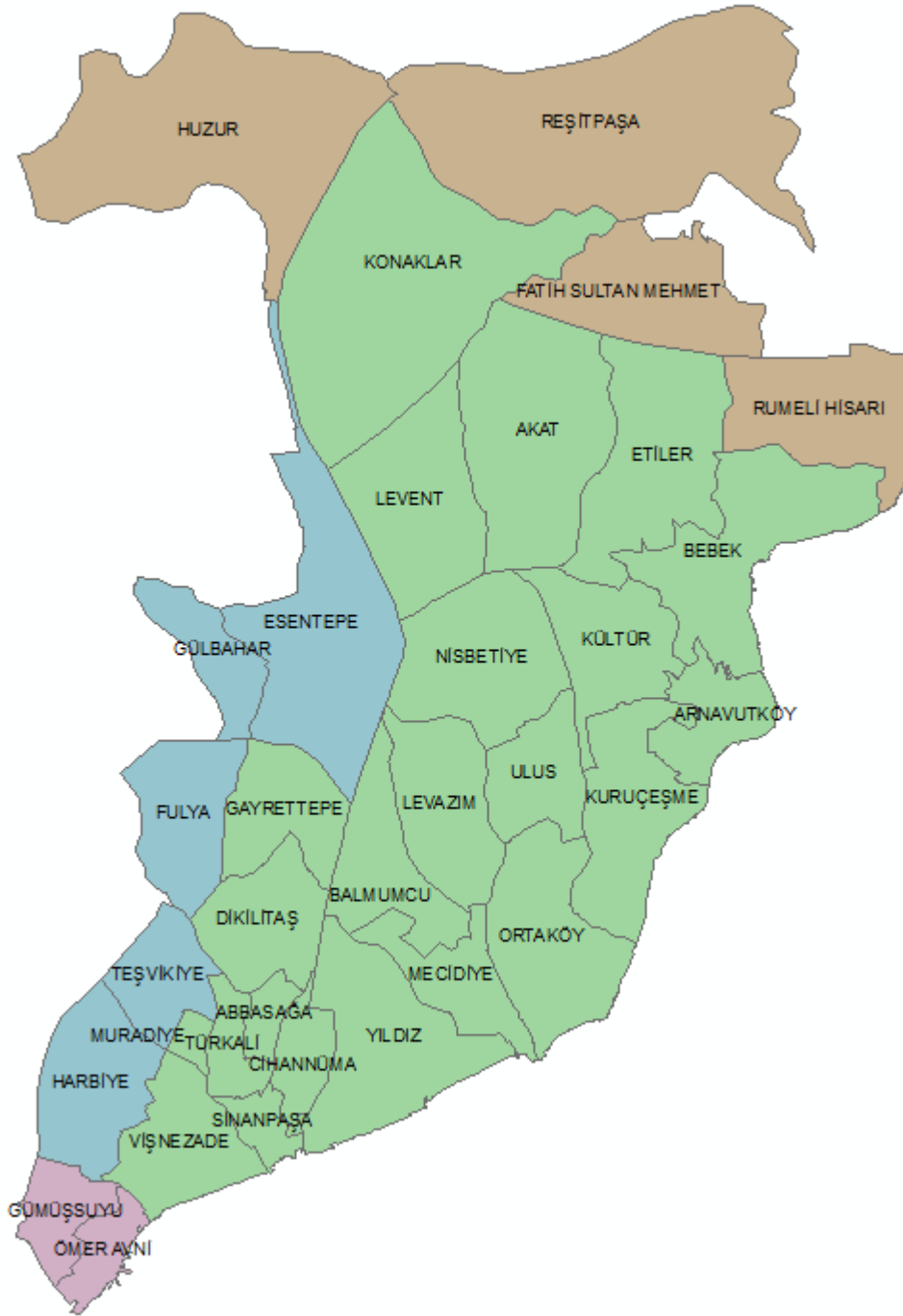
⁷⁶ <http://www.w3.org/RDF/Validator/>

4.3.1 Veri toplama ve analiz

Tez kapsamında kullanılacak veriler, İBB Planlama Müdürlüğünden GDB formatında edinilmiştir. Verilerin, önceki bölümde oluşturulan İdari Birimler Ontolojisi tanımlamalarına uygun olarak düzenlemesi, idari birimler arasındaki konumsal ilişkilerin ve idari birim geometrilerinin sorgulamalarla elde edilmesi gerekmektedir. Bu amaç doğrultusunda, edinilen veriler ArcGIS 10.2 masaüstü yazılımıyla düzenlenmeye ve sorgulanmaya başlanmıştır. İdari Birim Ontolojisinde, ülkemizdeki iki farklı idari birim modeli yapılandırılmıştır. Tez kapsamındaki uygulamanın, İstanbul İli, Beşiktaş İlçesinde yapılacak olması, bu modellerden 'Büyükşehir İl' sınıfının bulunduğu modeli uygulama olanağı vermiştir. 'Büyükşehir İl', 'Büyükşehir İlçe', 'Büyükşehir İl Belediyesi', 'Büyükşehir İlçe Belediyesi' ve 'Mahalle' sınıflarına uygun olarak verilerin toplanması için ArcGIS sorgulamaları yapılmıştır. İdari birimlerin topolojik ve mereolojik ilişkileri bu sorgulamalarla edinilmiştir. ArcGIS 'Select By Location' özelliği ve 'tamamen içindedir' sorgusu kullanılarak, Beşiktaş İlçesi içindeki mahalleler elde edilmiştir. Beşiktaş İlçesinin komşusu olduğu ilçeler ve mahalleleri belirlemek için, Beşiktaş ilçe sınırının diğer ilçe mahallelerinin sınırlarıyla kesiştiği yerlere nokta atılmıştır. Böylece, Beşiktaş ilçesinin komşu ilçeleri Sarıyer, Beyoğlu ve Şişli olarak belirlenmiştir. Ayrıca, bu ilçelerin hangi mahallelerinin Beşiktaş ile sınırı olduğu belirlenmiştir. Şekil 4.6'da, tez kapsamındaki uygulamanın sınırları, yani Beşiktaş İlçesi ve Beşiktaş İlçesinin komşu idari birimleri için hazırlanmış harita görülmektedir. Konumsal ilişkiler Ontolojisinden belirlenen 'içerir', 'içindedir' ve 'eşittir' nesne ilişkileri elde edildikten sonra, 'Select By Location' özelliği ve 'sınırları dokunur' sorgusu ile her mahalle için tek tek komşu mahalleler belirlenmiştir.

Konumsal ilişkilerden sonra geometrik tanımlamalar yapılmıştır. İdari birim koordinatlarını poligon olarak elde edebilmek için idari birim detayları önce çizgiye, sonra noktaya çevirilmiş ve bu noktalara iki boyutlu koordinat bilgisi atanmıştır. Bu koordinatlar coğrafi koordinatlara çevirilmiş ve ilgili idari birim verilerinin bulunduğu alana kaydedilmiştir. Beşiktaş İlçesinin, Beşiktaş mahallelerinin ve komşu ilçe ile mahallelerin iki boyutlu coğrafi koordinatları, ArcGIS yardımıyla, KML formatına dönüştürülmüştür. KML dokümanları elde edilen coğrafi koordinat verilerinden, GML tanımlamalarına uygun dokümanlar da oluşturulmuştur. Bu süreçte, veri tabanında

İstanbul’u ifade eden nokta sayısının fazlalığı nedeniyle, İstanbul il sınırının koordinatları genelleştirme yoluyla elde edilmiştir.



Şekil 4.6 Tez kapsamında belirlenen pilot bölge

Veriden elde edilen tüm sorgulama sonuçları; idari birim geometri, konumsal ilişki ve öznitelik verileri İdari Birim Ontolojisi sınıflarına, nesne ilişkilerine ('mahalle içerir', 'ilçe

İçerir', 'ilçe içindedir' vd.) ve veri ilişkilerine ('nüfus', 'posta Kodu', 'yüzölçüm' vd.) uygun olarak düzenlenmiştir. Coğrafi veri tabanından elde edilen öznitelikler Türkiye İstatistik Kurumu (TÜİK) verileri ile karşılaştırılmış ve güncel veriler kaydedilmiştir. Ayrıca, İBB'den elde edilen verilerde Şişli İlçesine bağlı görülen Huzur Mahallesinin, 2012 yılında alınan bir kararla Sarıyer Belediyesine bağlandığı da göz önünde tutularak, toplanan tüm güncel verilerle bir excel tablosu oluşturulmuştur.

Çalışma alanına ilişkin veriler İdari Birim Ontolojisi tanımlamalarına göre düzenlendikten sonra, verilerin hangi yöntemle Bağlı Veri olarak yayımlanacağına karar verilmesi gerekir. Çalışma alanının görece küçük olması, sınır verileri gibi temel nitelikteki bir verinin sıklıkla değişmemesi ve ilişkisel veri tabanlarından doğrudan Bağlı Veri yayımlarken karşılaşılan sorunlar nedeniyle; 'statik yöntemin' uygulanmasına karar verilmiştir. Verileri RDF dokümanları ile temsil etmeden önce, her idari birimi niteleyecek bir eşsiz tanımlayıcı belirlenmelidir. İzleyen bölüm, idari birimlere URI atamasına ilişkin metodolojiyi açıklamaktadır.

4.3.2 İdari Birimler için URI Tanımlama İlkeleri

Semantik Web gerçek dünya nesnelerini ve bunlarla ilgili dokümanları adlandırırken, tanımlayıcı olarak URI'lerin kullanılmasını önerir. Semantik Web'deki şey'leri adlandırmada kullanılan URI'lerle ilgili yaklaşımlardan ve URI modellerinden Bölüm 3.2'de söz edilmişti. Buna göre, URI oluşturmada iki yaklaşım uygulanmaktadır: (1) 303 URI ve (2) Hash URI yaklaşımları. Bu çalışmada, 303 URI'lerinin esnek bir tanımlama mekanizması sağlaması ve yapılan pilot çalışmanın daha sonra büyüyecek bir yapıda olması nedenlerinden dolayı 303 URI yaklaşımının kullanılmasına karar verilmiştir. Bu yaklaşımda; gerçek dünya nesnesi için bir URI, bu nesnenin Web sunumu için bir URI ve bu nesne ile ilgili bir Web dokümanı için de ayrı bir URI tanımlanması gerekir. Gerçek dünya nesnesinin URI'si yeniden referanslandırıldığında, '303 see other' yönlendirmesi ile diğer URI'lere yönlendirilmelidir. 303 URI yaklaşımının seçimi dışında, mantıksal ve kalıcı URI geliştirmek için bir modelinin geliştirilip, uygulanması gerekmektedir. Bunun için URI'lere açık olarak adların yazılması veya kalıcı başka bir tanımlayıcıyla URI'lerin şekillendirilmesi gibi iki yaklaşım bulunmaktadır. Çalışma kapsamında idari birim adlarının açık olarak URI'lerde yer almasına karar verilmiştir.

Çalışma kapsamında ontolojiler, gerçek dünya nesneleri, gerçek dünya nesneleriyle ilgili dokümanlar ve dokümanların Web sunumu için modeller kapsamında oluşturulan URI formatları Çizelge 4.3'te verilen biçimde tasarlanmıştır.

Tüm URI'lerin alan tanımlaması <http://araziyonetimi.org> URI'si ile yapılmıştır. Oluşturulan üç ontoloji de, ontoloji altdizininde toplanmış ve tanımlamaları kendi alanında yapılmıştır. Örneğin, İdari Birim Ontolojisindeki 'mahalle içerir' nesne ilişkisi <http://araziyonetimi.org/ontoloji/idaribirim/mahalleicerir> URI'si ile tanımlanmıştır. Ontoloji URI'leri belirlenirken, sınıf adlarının her kelimesinin ilk harfi büyük, ilişki adlarının ilk kelimesinin ilk harfi küçük diğer kelimelerinin ilk harfleri büyük olarak adlandırılarak URI'ler şekillendirilmiştir. İdari birimleri adlandırmada kullanılacak URI'leri oluştururken de Çizelge 4.3'teki diğer modeller kullanılmıştır. Her idari birim için altı farklı URI tanımlaması yapılmıştır. İdari birimlerin gerçek dünyadaki tanımlayıcıları olan URI'ler, Web sunumunun URI'leri ve RDF dokümanlarının URI'leri bunların ilk üçüdür. Diğer üç URI ise idari birimlerin gerçek dünyadaki geometrisinin URI'si, Web sunumunun geometrisinin URI'si ve RDF dokümanında ifade edilen geometrinin URI'sidir. Gerçek dünya nesnelerini ifade ederken 'id', bunlara ilişkin Web sunumunu ifade ederken 'doc' altdizinleri kullanılmıştır. RDF dokümanlarının URI'si ise 'rdf' uzantısı ile belirlenmiştir. Tüm il, ilçe, mahalle URI'lerinin oluşturulmasında bu modele bağlı kalınmıştır. Örneğin, Abbasağa mahallesinin gerçek dünyadaki tanımlayıcısı <http://araziyonetimi.org/id/idaribirim/istanbul-besiktas-abbasaga> şeklinde belirlenmiştir. Büyükşehir belediyeleri ve büyükşehir ilçe belediyelerini adlandırırken de aynı URI'lerin sununa 'bb' ve 'b' kısaltmaları eklenmiştir. Örneğin, <http://araziyonetimi.org/id/idaribirim/istanbulbb-besiktasb> gibi. Belirlenen bu model, URI'lerde açıkça neyin adlandırıldığını görmeye olanak tanıyan bir adlandırma modelidir. Bu model; kullanılarak ülkemizdeki tüm idari birimler eşsiz bir biçimde adlandırılabilir ve URI'yi yeniden referanslandırmadan, doğrudan URI adından, URI'nin ne hakkında olduğu anlaşılabilir olduğu için seçilmiştir. Sonuç olarak bu bölümde, çalışma alanındaki tüm idari birimler ve bunların geometrileri için URI'ler hazırlanmıştır. Hazırlanan URI'ler, RDF dokümanlarını oluşturma aşamasında kullanılacaktır. RDF dokümanlarının hazırlanmasında kullanılan ontolojiler, ilişkiler ve RDF dokümanlarının yapısı aşağıda anlatılmaktadır.

Çizelge 4.3 Çalışma kapsamında kullanılan URI yapıları

Ontoloji URI	http://araziyonetimi.org/ontoloji/idaribirim
	http://araziyonetimi.org/ontoloji/konumsaliliskiler
	http://araziyonetimi.org/ontoloji/geometri
Gerçek dünya nesneleri URI	http://araziyonetimi.org/id/idaribirim/iladi-ilceadi-mahalleadi
Geometri gerçek dünya nesneleri URI	http://araziyonetimi.org/id/idaribirim/geometri/iladi-ilceadi-mahalleadi
Web URI	http://araziyonetimi.org/doc/idaribirim/iladi-ilceadi-mahalleadi
Geometri Web URI	http://araziyonetimi.org/doc/idaribirim/geometri/iladi-ilceadi-mahalleadi
Doküman URI	http://araziyonetimi.org/doc/idaribirim/iladi-ilceadi-mahalleadi.rdf
Geometri Doküman URI	http://araziyonetimi.org/doc/idaribirim/geometri/iladi-ilceadi-mahalleadi.rdf

4.3.3 İstanbul İli Beşiktaş İlçesi RDF Dokümanlarının Oluşturulması

Önceki bölümde açıklandığı gibi, çalışma alanındaki her bir idari birim ve bunların geometrileri için URI'ler oluşturulmuştur. Bu bölümde, elde edilen veriler ve URI tanımlamaları kullanılarak, çalışma alanındaki her bir idari birim ve bunların geometrileri için RDF dokümanları oluşturulacaktır. RDF dokümanlarının oluşturulması aşamasında 'Google Refine RDF Extension' yazılımından faydalanılmıştır. Yazılımın ürettiği RDF dokümanlarına bazı durumlarda elle müdahale gerekmiştir.

Çalışma alanı incelendiğinde beş farklı idari birim yapısının olduğu görülmüştür. 'Büyükşehir İl', 'Büyükşehir İl Belediyesi', 'Büyükşehir İlçe', 'Büyükşehir İlçe Belediyesi' ve 'Mahalle' için, İdari Birim Ontolojisi sınıflarında yapılan tanımlamalara göre beş farklı RDF modeli oluşturulmuştur. RDF modellerinde; İdari Birim, Geometri ve Konumsal İlişkiler Ontolojileri tanımlamalarının yanı sıra, diğer ontolojiler ile sunulan tanımlamalardan da yararlanılmıştır. Örneğin, alternatif adları ifade etmek için SKOS, nokta koordinatları ifade etmek için WGS84 ve GeorSS, meta verileri ifade etmek için 'Dublin Core Metadata' ve lisanslama bilgilerini ifade etmek için 'Creative Commons' ontolojileri kullanılmıştır. RDF dokümanları hakkında ayrıntılı bilgi Bölüm 3.3'te verilmiştir.

Çalışma kapsamında 44 farklı idari birim için, 88 RDF dokümanı oluşturulmuştur. Her idari birimin öznitelikleri ve konumsal ilişkilerini içeren bir RDF dokümanı ile idari birimlerin geometrilerinin poligon olarak ifade edildiği ikinci bir RDF dokümanı, hem RDF/XML hem de Turtle formatında hazırlanmıştır. Çizelge 4.4'te Abbasağa Mahallesi için hazırlanan RDF/XML formatındaki dokümandan bir parça görülmektedir. Her RDF dokümanı, özelleşmiş bir XML dokümanı olduğundan, her RDF bunu ifade eden bir ilk satır ile başlar. Daha sonra, dokümanda RDF tanımlamalarının yapılacağı rdf:RDF özelliği ile belirtilir. Ardından, kullanılacak ontolojilerin tanımlamalarının bulunduğu URI'ler ad alanları ile birlikte verilir ve üçlüler halinde RDF ifadeleri oluşturmaya başlanır. Örnekte ilk belirtilen RDF ifadesi Abbasağa mahallesinin posta kodudur. Daha sonra Abbasağa Mahallesi'nin, Beşiktaş Belediyesi'nin ve İBB'nin içinde olduğu, iki ayrı ifade ile belirtilmiştir. Abbasağa Mahallesi'nin Vikipedi adresi, Web dokümanının hangi URI'de olduğu, UAVT kodu ve enlemi sırasıyla RDF ifadeleri olarak belirlenmiştir.

Tüm RDF/XML ve Turtle formatındaki dokümanlar, Jena RDF API ile yapısal olarak doğrulanmıştır. Yapısal doğrulama, W3C online doğrulama servisi ve Bölüm 3.3.2’de sözü edilen diğer servislerle yapılabilmektedir. RDF dokümanlarının ontolojilerdeki tüm tanımlamalara uygun üretilip üretilmediğini denetleyen ve yaygın olarak kabul edilmiş bir araç yoktur. Bu nedenle, RDF dokümanlarının sadece yapısal doğruluğu denetlenmiştir. Çalışma kapsamında oluşturulmuş RDF/XML ve Turtle formatındaki dokümanlardan birer örnek EK D’de verilmiştir.

Çizelge 4.4 İdari Birim RDF/XML örneği

```
<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF
  xmlns:idaribirim="http://araziyonetimi.org/ontoloji/idaribirim/"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:wgs84_pos="http://www.w3.org/2003/01/geo/wgs84_pos#"
  <idaribirim:postaKodu>34353</idaribirim:postaKodu>
  <idaribirim:belediyeKodu rdf:resource="http://araziyonetimi.org/id/idaribirim/istanbulbb-besiktasb"/>
  <idaribirim:belediyeKodu rdf:resource="http://araziyonetimi.org/id/idaribirim/istanbulbb"/>
  <idaribirim:wikipedi rdf:resource="http://tr.wikipedia.org/wiki/Abbasa%C4%9Fa,_Be%C5%9Fikta%C5%9F"/>
  <rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/id/idaribirim/istanbul-besiktas-abbasaga.rdf"/>
  <idaribirim:UAVTKodu>40219</idaribirim:UAVTKodu>
  <wgs84_pos:lat>41.05125</wgs84_pos:lat>
```

Çalışmanın bu aşamasına dek Bağlı Veri yaklaşımının ilk üç ilkesi gerçekleştirilmiştir: İdari birimler HTTP URI’lerle adlandırılmış, coğrafi veri tabanından elde edilen veriler ile diğer öznitelik verileri, geliştirilen ontolojilerle RDF veri modeline aktarılmıştır. Bağlı Veri yaklaşımının dördüncü ilkesi ise tanımlanan nesnelerin diğer nesnelerle anlamsal ilişkilerle bağlanmasını öngörür. Bölüm 3’te RDF dokümanları arasında bağlantılar kurulmasına olanak sağlayan DBpedia, Freebase Geonames gibi bilgi tabanlarından (knowledge base) söz edilmmişti. Bu bilgi tabanları, Bölüm 3.6’da belirtildiği gibi, çeşitli yazılımlarla RDF’ler arasında manuel, yarı otomatik veya otomatik bağlantılar kurulmasına olarak sağlar. Çalışma görece küçük bir alanda yapıldığı için bağlantıların kurulmasında SILK gibi otomatik eşleşme sağlayan bir yazılım yerine, yarı otomatik eşleme sağlayan ‘Google Refine’ ile semantik arama motorları aracılığıyla yapılan aramalardan yararlanılmıştır. Çalışma alanındaki mahallelerin birçoğunun Web’de yayımlanmış bir benzeri bulunamamıştır.

Birbirleriyle anlamsal ilişkiler içeren RDF’ler, owl:sameAs ve rdfs:seeAlso özellikleri ile ilişkilendirilebilir. owl:sameAs, RDF dokümanlarının tamamen birbirinin aynısı olduğunu

ifade ettiğinden dolayı, daha zayıf bir bağlantı kuran `rdfs:seeAlso` özelliği uygulanmıştır. Örneğin, Beşiktaş ilçesi için DBpedia’da oluşturulan <http://dbpedia.org/page/Be%C5%9Fikta%C5%9F> ve Geonames’de oluşturulan <http://www.geonames.org/7733234/> URI’leri, bu çalışma kapsamında Beşiktaş idari birimi için geliştirilen <http://araziyonetimi.org/id/idaribirim/istanbul-besiktas> URI’si, `rdfs:seeAlso` özelliği ile ilişkilendirilmiştir. İzleyen bölüm, hazırlanan RDF dokümanlarının Bağlı Veri olarak yayımlanmasına ilişkin süreci özetlemektedir.

4.3.4 RDF Dokümanlarının Bağlı Veri Olarak Yayımlanması

Bu aşamada RDF dokümanları, bir RDF veri tabanı (triple store) kullanılarak saklanacak ve Bağlı Veri olarak yayımlanacaktır. Bağlı Veri yayımlayan ve yayımlanan verilerin SPARQL ile sorgulanmasına olanak sağlayan birçok farklı araç bulunmaktadır (bkz. Bölüm 3.7). Bu araçlar içinde Fuseki ve Virtuoso en çok tercih edilen yazılımlardır. Virtuoso, sınırsız RDF ifadesi sunabilme ve RDF veri setlerine yetkilendirme sağlayabilme özellikleri ile Fuseki’nin önüne geçmektedir. Fakat kullanımının görece Virtuoso’dan kolay olması nedeniyle Bağlı Veri yayımlamada Fuseki tercih edilmiştir. Fuseki, on milyona kadar RDF ifadesinin ve HTTP protokolleri üzerinden RDF dokümanlarının sunulmasına olanak sağlar. Bu aşamada izlenen süreç kısaca şöyledir: Gerekli düzenlemeler yapıldıktan sonra, Fuseki <http://localhost:3030/> adresinde lokal olarak çalışır duruma getirilir. Çalışma alanındaki idari birimler için üretilen RDF dokümanları, HTTP komutları ile Fuseki’ye yüklenir. Fuseki, Jena tarafından geliştirilen RDF saklama aracı olan TDB¹ yazılımının üzerine kurulmuştur ve Fuseki’ye yüklenen veriler burada saklanır. Fuseki’ye RDF dokümanı yüklemek için HTTP ‘put’ komutu, dokümanı geri çağırmak için ‘get’ komutu, dokümanı sorgulayabilmek için ‘query’ komutu ve güncelleme için ‘update’ komutu kullanılır. Şekil 4.7’de Abbasağa Mahallesi hakkındaki RDF dokümanının ‘put’ komutu ile Fuseki üzerinden TDB’ye yüklendiği görülmektedir.

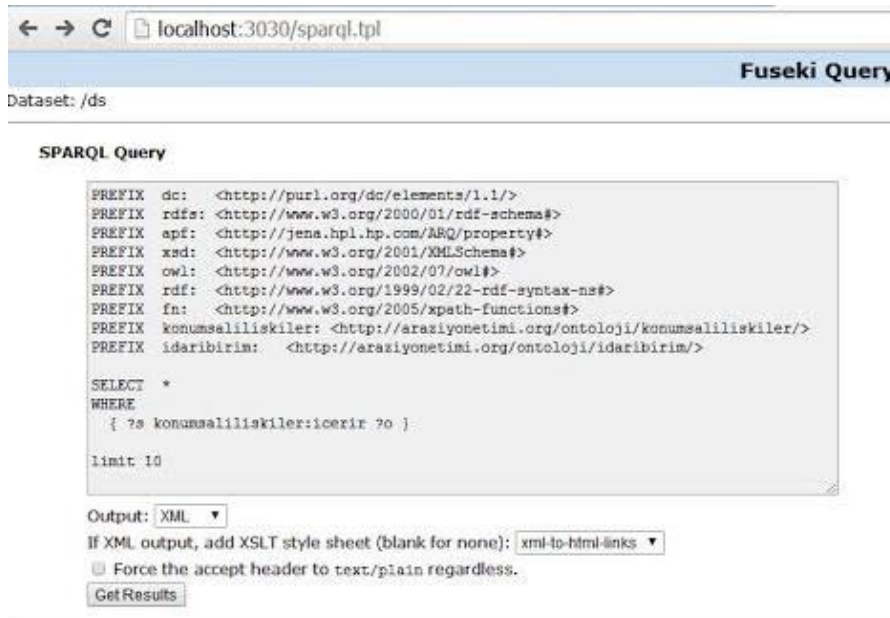
¹ <https://jena.apache.org/documentation/tdb/index.html>



```
Administrator: Start Command Prompt with Ruby
ruby 2.0.0p481 (2014-05-08) [i386-mingw32]
C:\Users\Administrator>cd C:\temp\jena-2.12.0\jena-fuseki
C:\temp\jena-2.12.0\jena-fuseki>ruby s-put http://localhost:3030/ds/data default
istanbul-besiktas-abbasaga.ttl
C:\temp\jena-2.12.0\jena-fuseki>
```

Şekil 4.7 Fuseki'ye veri yükleme

İdari birimler hakkında üretilen tüm RDF dokümanları TDB'de saklandıktan sonra, veriler üzerinde Fuseki aracılığıyla SPARQL sorgulamaları yapılabilir. RDF veri tabanında saklanan veriler, Fuseki aracılığıyla <http://localhost:3030/ds/query> üzerinde SPARQL Endpoint olarak yayımlanmıştır ve lokal olarak SPARQL sorgulamalarına yanıt verebilir durumdadır. SPARQL Endpoint, arayüzlerin SPARQL ile sorgulanmasına olanak tanır. Şekil 4.8'de idari birim RDF dokümanları üzerinde yapılan bir SPARQL sorgusu görülmektedir. Bu sorguda, öznesi ve nesnesi Konumsal İlişkiler Ontolojisinde tanımlanan 'içerir' nesne ilişkisi ile birbirine bağlanmış tüm RDF ifadeleri sorgulanmaktadır. 'limit 10' ifadesi ile de, elde edilen sonuçlardan sadece on tanesinin gösterilmesi istenmiştir.



localhost:3030/sparql.tpl

Fuseki Query

Dataset: /ds

SPARQL Query

```
PREFIX dc: <http://purl.org/dc/elements/1.1/>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX apf: <http://jena.hpl.hp.com/ARQ/property#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX fn: <http://www.w3.org/2005/xpath-functions#>
PREFIX konumsaliliskiler: <http://araziyonetimi.org/ontoloji/konumsaliliskiler/>
PREFIX idaribirim: <http://araziyonetimi.org/ontoloji/idaribirim/>

SELECT *
WHERE
{ ?s konumsaliliskiler:içerir ?o }

limit 10
```

Output: XML

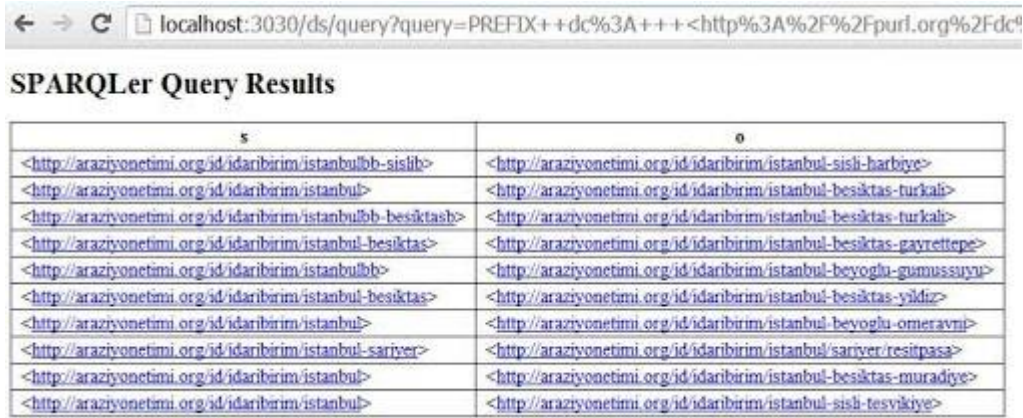
If XML output, add XSLT style sheet (blank for none): xsl-to-html-links

☐ Force the accept header to text/plain regardless.

Get Results

Şekil 4.8 Fuseki ile idari birim RDF'lerinin sorgulanması

İdari birim RDF dokümanlarında oluşturulan RDF ifadelerine göre, öznesi ve nesnesi arasında 'içerir' nesne ilişkisi bulunan, on adet idari birim Şekil 4.9'da görüldüğü gibidir. Lokalde yayımlanan, idari birimler hakkındaki SPARQL Endpoint'ler, yeniden yazım kuralları (rewrite rule) ile <http://araziyonetimi.org/sparql> adresine yönlendirilmiş ve idari birim RDF'leri başka servisler tarafından sorgulanabilir konuma getirilmiştir. EK E'de Fuseki kullanılarak yapılan farklı sorgulamalara yer verilmiştir.



SPARQLer Query Results

s	o
< http://araziyonetimi.org/id/idaribirim/istanbulb-sislib >	< http://araziyonetimi.org/id/idaribirim/istanbul-sisli-harbiye >
< http://araziyonetimi.org/id/idaribirim/istanbul >	< http://araziyonetimi.org/id/idaribirim/istanbul-besiktas-turkali >
< http://araziyonetimi.org/id/idaribirim/istanbulb-besiktas >	< http://araziyonetimi.org/id/idaribirim/istanbul-besiktas-turkali >
< http://araziyonetimi.org/id/idaribirim/istanbul-besiktas >	< http://araziyonetimi.org/id/idaribirim/istanbul-besiktas-gayrettepe >
< http://araziyonetimi.org/id/idaribirim/istanbulb >	< http://araziyonetimi.org/id/idaribirim/istanbul-beyoglu-gumussuyu >
< http://araziyonetimi.org/id/idaribirim/istanbul-besiktas >	< http://araziyonetimi.org/id/idaribirim/istanbul-besiktas-yildiz >
< http://araziyonetimi.org/id/idaribirim/istanbul >	< http://araziyonetimi.org/id/idaribirim/istanbul-bevoglu-omeravni >
< http://araziyonetimi.org/id/idaribirim/istanbul-sariyer >	< http://araziyonetimi.org/id/idaribirim/istanbul-sariyer-resitpasa >
< http://araziyonetimi.org/id/idaribirim/istanbul >	< http://araziyonetimi.org/id/idaribirim/istanbul-besiktas-muradiye >
< http://araziyonetimi.org/id/idaribirim/istanbul >	< http://araziyonetimi.org/id/idaribirim/istanbul-sisli-tesvikiye >

Şekil 4.9 İdari birim dizini Fuseki sorgulama sonucu

Bu aşamada, çalışma kapsamında üretilen idari birim örneklemeleri; diğer benzer çalışmalarla bağlantılı hale getirilmiş, Bağlı Veri olarak yayımlanmış ve ayrıca, SPARQL Endpoint olarak sunulurken, sorulamaya hazır hale getirilmiştir. Böylece, Semantik Web'de uygulama geliştirecek bir yazılımcı, sunulan idari birim verilerinden sorgulamalarla elde ettiği URI'leri, doğrudan uygulamalarında kullanabilecektir.

Çalışma kapsamında yayımlanan idari birim verilerinin keşfedilebilir olmasını sağlamak için, VOID, 'Semantik Sitemap' gibi sözlükler kullanılarak, Bağlı Veri olarak yayımlanmış veri setleri hakkındaki verilerin (metadata) yayımlanması yoluyla, verilerin keşfedilebilirliği artırılır. Çalışma kapsamında sunulan idari birim veri seti hakkında veriler sunmak için, bir VOID RDF sözlüğü kullanılarak, bir RDF dokümanı hazırlanmıştır [EK E]. RDF dokümanında idari birim örneklemeleri oluşturulurken kullanılan sözlükler, veri setinin lisans bilgileri ve Web sunumuyla ilgili bilgileri içermektedir. Yine, yayımlanan Bağlı Verilerin keşfedilebilirliğini arttırmak için, Web sunumu yapılması önerilmektedir. Sonraki bölümde, tez kapsamında geliştirilen idari birim Bağlı Veri

setinin keşfedilebilirliğini arttırmak için hazırlanmış bir Web sitesi ve sitenin içeriği anlatılmıştır.

4.3.5 Türkiye İdari Birimler Dizininin İstanbul İli Beşiktaş İlçesi Bağlı Veri Uygulaması ve Web Sunumu

İdari Birim Ontolojisi, Konumsal İlişkiler Ontolojisi ve Geometri Ontolojisi ile çalışma alanındaki idari birimleri ifade eden veriler <http://araziyonetimi.org/> adresinde yayımlanan web sitesi ile kullanıcılara sunulmuştur. Web sitesinin ‘Ontoloji’ dizininden çalışma kapsamında hazırlanan üç ontolojiye (Şekil 4.10); ‘Liste’ dizininden ise idari birimlerin RDF dokümanlarına erişilebilir (Şekil 4.11). Ayrıca, ‘Sorgu’ dizininde yapılan aramalarla, idari birim örneklemelerine erişilebilir. Sorgu bölümünde SPARQL ile sorgulama olanağı da sağlanmıştır.



Şekil 4.10 Türkiye idari birimler dizini Web sitesi ontoloji bölümü

Alfabetik İdari Birimler Listesi

- [İstanbul](#)
- [İstanbul Büyükşehir Belediyesi](#)
- [Beşiktaş](#)
- [Beşiktaş Belediyesi](#)
- [BeYOĞLU](#)
- [BeYOĞLU Belediyesi](#)
- [Sarıyer](#)
- [Sarıyer Belediyesi](#)
- [Şişli](#)
- [Şişli Belediyesi](#)
- [Abbasağa](#)
- [Akat](#)
- [Arnavutköy](#)
- [Balmumcu](#)
- [Bebek](#)
- [Cihannuma](#)
- [Dikilitaş](#)
- [Esentepe](#)
- [Etiler](#)
- [Fatih Sultan Mehmet](#)
- [Fulya](#)
- [Gavrettepe](#)
- [Gülbahar](#)
- [Gümüşsuyu](#)
- [Harbiye](#)
- [Huzur](#)
- [Konaklar](#)
- [Kültür](#)
- [Kuruçeşme](#)
- [Levazım](#)
- [Levent](#)
- [Mecidiye](#)
- [Muradiye](#)
- [Nispetiye](#)
- [Ömer Avni](#)
- [Ortaköy](#)
- [Reşitpaşa](#)
- [Rumeli Hisarı](#)
- [Sinanpaşa](#)
- [Teşvikiye](#)
- [Türkali](#)
- [Ulus](#)
- [Vişnezade](#)
- [Yıldız](#)

Şekil 4.11 Türkiye idari birimler dizini Web sitesi liste bölümü

İdari Birim Yer Adları Dizininin Bağlı Veri olarak yayımlanıp yayımlanmadığı, 'Vapour' servisi ile kontrol edilmiştir. Örneğin, Şekil 4.12'de Abbasağa Mahallesinin Bağlı Veri olarak yayımlandığının doğrulanması görülmektedir.

The screenshot displays the 'Validate by URI' section of the Vapour Report. It features a text input field for a URI, with the example 'http://araziyonetimi.org/doc/idaribirim/istanbul-besiktas-abbasaga' entered. Below the input field is a 'CHECK' button. A green banner below the input field states 'All tests passed!'. Underneath, a 'Summary:' section contains a table with two columns: 'Test requirement' and 'Passed tests'.

Test requirement	Passed tests
Dereferencing resource URI (requesting RDF/XML)	2/2
Dereferencing resource URI (without content negotiation)	1/1

Şekil 4.12 İdari birim Web sunumlarının Bağlı Veri olduğunu doğrulaması

Her idari birimin sayfasında RDF'den elde edilen bilgiler gösterilmiştir. İdari birimlerin geometrileri, RDF dokümanlarından GML tanımlamlarına uygun olarak sunulmuş, ayrıca, KML verileri ile hazırlanan haritalar da her idari birimin sayfasında sunulmuştur. Şekil 4.13'te Abbasağa Mahallesi için hazırlanan Web sayfası görülmektedir.

HAKKIMIZDA

```

Geometri
gml:Olarak
srcName="Polygon xmlns:gml="http://www.opengis.net/gml"
srsName="http://www.opengis.net/def/crs/EPSG/0/38571"
<gml:exterior>
<gml:LinearRing>
<gml:posList srsDimension="2">
29.007800448 41.05109858
29.00782373 41.05095622
29.00743901 41.05065323
29.00727839 41.05053768
29.0072307 41.05046629
29.00684895 41.05006711
29.00664234
41.04981696
29.00652606 41.04965238
29.00647741 41.04954303
29.00645451
41.04940969
29.00636641 41.04931938
29.00622404 41.04899856
41.04886198
29.00601819 41.04821763
29.00593179 41.04789782
29.00589652
41.04776946
29.00578667 41.04738882
29.00576125 41.04729874
29.00573818
41.04716727
29.00553298 41.04673849
29.00505141 41.04675952
29.00478821
41.04682542
29.00460369 41.04686849
29.00441766 41.0468934
29.00439067
41.04691752
29.00436128 41.0469331
29.00432217 41.04694178
29.00428046
41.04693731
29.00421007 41.04715086
29.00370291 41.04701748
29.00339695
41.04690669
29.00336623 41.04689562
29.00329282 41.04693734
29.00327948
41.04702502
29.0033826 41.04713667
29.00323925 41.04735701
29.00323745
41.04742264
29.0032322 41.04764271
29.00319658 41.04800382
29.00319956
41.04816193
29.00320818 41.04831307
29.00323647 41.04855808
29.00326442
41.04873039
29.00328071 41.0488375
29.00326677 41.04899855
29.00324696
41.04907626
29.00321894 41.04920879
29.00321262 41.04930964
29.00323551
41.04941452
29.00327287 41.04848797
29.00331976 41.04855547
29.00333504
41.04959519
29.00337533 41.04963009
29.00341784 41.04969696
29.00344968
41.04974808
29.00349381 41.04977865
29.00362454 41.04984822
29.00381813
41.04994342
29.00387751 41.04995824
29.00392647 41.0499751
29.00384358
41.05021908
29.00327287 41.05068016
29.00370101 41.05081988
29.00365716
41.05090957
29.0037203 41.05100534
29.00386248 41.05108914
29.00393677
41.05112552
29.00343541 41.05129202
29.00452791 41.05140971
29.00488341
41.05162081
29.00508612 41.05179506
29.00537626 41.05201601
29.00506876
41.0522094
29.00516649 41.0521418
29.0050505
41.0523564
29.00512671
41.05239331
29.006112
41.05244477
29.00626715 41.05245885
29.0064039
41.05246592
29.00647286 41.05227921
29.00650928 41.05212742
29.00652781
41.05202553
29.0065859 41.05191471
29.00665094 41.05181647
29.00675156
41.05177862
29.00686261 41.05174474
29.00702231 41.05170664
41.00718542
41.05165245
29.00721333 41.05162479
29.00745263 41.05147369
29.00777442
41.05127891
29.0078213 41.05123578
29.00779963 41.05120192
29.00800448
41.05109858
</gml:posList>
</gml:LinearRing>
</gml:exterior>
</gml:Polygon>
enlem
41.05125
boylam
29.00555

```

SONUÇ VE ÖNERİLER

Sayısal yer adları dizinleri, konumla ilişkilendirilmiş yer adlarına odaklanarak Web temelli haritacılık servislerinin temelini oluşturur. Küresel yer adları dizinleri, günümüzde Semantik Web yaklaşımı ve Bağlı Veri teknolojileri ile yayımlanmaya başlamıştır. Bu yaklaşım ve teknolojiler, farklı Web servislerini ve/veya farklı veri formatlarını kullanan dizinlerin tek bir sorgu ile aynı anda sorgulanabilmesine olanak sağlar. Ayrıca, Bağlı Veri yaklaşımı, farklı alanlardaki ve formatlardaki veriler arasında ilişkileri tanımlayarak keşfedilebilir ve sorgulanabilir büyük bir veri uzayı oluşturur.

Bu çalışma, yer adları dizinlerinin idari birim detayına ülkemiz ölçeğinde odaklanmaktadır. Çalışmada idari birim detayının seçilmesinin nedeni, idari birimlerin diğer veri temaları (örn., adres, kadastral parsel vd.) için referans olmasıdır.

İdari birim detayını içeren küresel yer adları dizinleri ve literatür, dizinlerin UML sınıf diyagramlarıyla veya ontolojilerle modellendiği örnekler sunmaktadır. Bu iki farklı model, temelde iki farklı yaklaşımı işaret etmektedir. Bunlardan ilki, ISO/TC 211 standartlarına dayalı biçimde veri tanımlamalarının UML sınıf diyagramlarıyla temsil edildiği Servis Yönelimli Mimari yaklaşımı; ikincisi ise veri tanımlamalarının ontolojilerle yapıldığı ve verilerin Bağlı Veri metodolojisi ile RDF ile temsil edilerek yayımlandığı Semantik Web yaklaşımıdır.

ISO/TC 211 standartlarına dayalı olarak geliştirilen Servis Yönelimli Mimari yaklaşımında, tanımlamalar UML sınıf diyagramları ile yapılır. Ancak, UML sınıf diyagramlarının mantıksal çıkarsamalara olanak vermemesi; çoklu miraslamayı (inheritence) ifade edememesi; öznelik tanımlamalarının sınıflara bağlı olarak olması

nedeniyle özniteliklerin tek başına bir anlam ifade etmemesi ve aynı kavramlar için farklı kişi veya kurumların yaptığı tanımlamalar arasında bağlantılar kurulmasına olanak vermemesi bu yaklaşımın başlıca eksiklikleridir.

İlk bakışta, Bağlı Veri ve ISO/TC 211 coğrafi bilgi standartlarının; konumsal veriye erişimi, birlikte çalışabilirliği, verilerin entegrasyonunu ve kullanımını kolaylaştırmayı amaçlayan aynı temel ilkeyi paylaştıkları söylenebilir [17, 125]. Fakat Bağlı Veri yaklaşımının, bazı üstünlükleri vardır: Bunlardan ilki veri tanımlamalarının yapıldığı ontolojilerin kullanımıdır. Ontolojiler, mantıksal çıkarım yapılmasına olanak sağlamakta ve çoklu hiyerarşik ilişkilere olanak vermektedir. Bunların yanında, UML sınıf diyagramlarından farklı olarak, bir ontoloji ile benzer tanımlamalar içeren ontolojilerin sınıfları, ilişkileri ve öznitelikleri arasında bağlantılar kurulabilmektedir. Ontolojiler ve Bağlı Veri metodolojisiyle yayımlanan RDF dokümanları Web’de platformdan bağımsız biçimde yayımlanır. Böylelikle, farklı alanlardaki (örn., idari birim ve kadastral parsel) verilerin entegrasyonunu etkinleştirir. Aynı coğrafi detaya ilişkin veri üreten kuruluşların, bu verileri ontolojiler ile modelleyerek ve RDF ile temsil ederek Web ortamında yayımlaması, kullanıcıların tek bir sorgulama ile aynı detay ile ilgili tüm kaynaklara (örn., bilgi ve belgeler) Web üzerinden erişimini sağlar. Konumsal ve konumsal olmayan verilerin Web ortamında esnek biçimde paylaşımı, bilgiye erişim için büyük bir potansiyel taşır. Örneğin, Türkiye İstatistik Kurumu, Tapu ve Kadastro Genel Müdürlüğü, Gelirler Genel Müdürlüğü vd. kamu kuruluşlarınca aynı taşınmaz veya parsel için üretilen farklı veri setlerinin RDF temsili, kullanıcıların o taşınmaz / parsel ile ilgili tüm bilgi ve belgelere Web yoluyla erişimini olanaklı kılar. Bağlı Veri uzayında, aynı alan hakkında üretilen veri setlerinden kullanıcının istediği veriyi seçip uygulamalarında kullanabilmesi Bağlı Veri yaklaşımının bir diğer üstünlüğüdür.

Bağlı Veri yaklaşımının ISO/TC 211 standartlarına dayalı yaklaşımdan zayıf olduğu bazı alanlar da bulunmaktadır. Bağlı Veri sorgulama aracı olan SPARQL, coğrafi veri standartları ile şekillendirilmiş bir CBS yazılımı kadar etkin konumsal sorgulama araçları sunmamaktadır. Bu nedenle, birleşik konumsal sorgulamalarda CBS yazılımları, SPARQL’den daha etkili sonuçlara ulaşmaktadır. Ayrıca, çok büyük boyuttaki koordinat verilerinin saklanması veri tabanlarının, RDF dokümanlarından daha etkili olduğu bilinmektedir.

ISO/TC 211 standartlarına dayalı ilk yaklaşımın konumsal sektördeki uygulamasına INSPIRE projesi örnek olarak verilebilir. Avrupa Konumsal Veri Altyapısını oluşturmayı amaçlayan INSPIRE projesine uyum kapsamında yürütölmüş olan TUCBS projesi de, bu yaklaşımın ölkemizdeki uygulamasıdır. Bu tezin çalışma alanını oluşturan yer adları, TUCBS Adres Veri Temasında; idari birimler ise İdari Birimler Veri Temasında UML sınıf diyagramları ile modellenmiştir [123, 124]. Bu çalışma, gerçekleşmesi hedeflenen TUCBS projesini Bağlı Veri yaklaşımıyla desteklemeye odaklıdır.

Çalışmada, idari birimlere ilişkin mevzuatdan, literatürden ve TUCBS projesinden elde edilen verilerle Türkiye İdari Birimler Dizini için bir İdari Birimler Ontolojisi oluşturulmuştur. Farklı alanlardaki terimlerin farklı ontolojilerle modellenmesi gerektiğinden; idari birimlerin geometrilerini ifade etmek için Geometri Ontolojisi, idari birimler arasındaki konumsal ilişkileri ifade etmek için ise Konumsal İlişkiler Ontolojisi hazırlanmıştır. Oluşturulan üç farklı ontoloji de, SKOS ve RDFS ontoloji dillerine göre daha ayrıntılı tanımlamama yapma imkanı sunan ve mantıksal çıkarımlara izin veren OWL ontoloji dili ile modellenmiştir. Tezde geliştirilen üç ontoloji; idari birimlerin adları, konumları, öznitelikleri, geometrileri ve birimler arasındaki konumsal ilişkiler (mereolojik ve topolojik ilişkiler) formel biçimde tanımlanmasına olanak vermektedir. Ontoloji mühendisliği, daha önce yapılan benzer çalışmalarla bağlantılar kurulmasını öngördüğünden; Geometri ve Konumsal İlişkiler Ontolojileri ile tanımlanan ilişki ve öznitelikler, standart bir sözlük olan OGC GeoSPARQL ile eşleştirilmiştir. Örneğın, GeoSPARQL ‘contains’ ilişkisi ile Konumsal İlişkiler Ontolojisi ‘içerir’ ilişkisi birbirine eşitlenmiştir.

Tez kapsamında, İstanbul İli Beşiktaş İlçesi ve Beşiktaş İlçesi ve komşu idari birimlerine ilişkin RDF dokümanları hazırlanarak Semantik Web kapsamında Bağlı Veri olarak yayımlanmıştır. Bu kapsamda, idari birimlere eşsiz URI atamaları yapılmış, ontoloji tanımlamalarından yararlanarak idari birimler için RDF dokümanları hazırlanmıştır.

Verilerin Bağlı Veri olarak yayımlanması amacıyla geliştirilmiş türlü yaklaşımlar vardır. Bunlardan en çok tercih edileni veri tabanlarının Bağlı Veri formatına dönüşümüdür. Ancak bu yaklaşımın henüz standartlaşmış bir metodolojiye sahip olmaması ve veri tabanlarının anlamsallığını zedelemeyen RDF’ye dönüşüm güçlüklerinden ötürü;

çalışmada statik yöntem uygulanmıştır. Her idari birim detayı için hazırlanan RDF dokümanları, benzer çalışmalarla ve HTML görünülerıyla eşleştirilmiş, Statik Bağlı Veri olarak yayımlanmıştır. Statik yöntemle yayımlanan RDF dokümanlarının, Semantik Web sorgulama dili olan SPARQL ile sorgulanabilmesine olanak sağlamak için Apache Jena Fuseki yazılımından faydalanılmıştır. Fuseki yazılımı ile SPARQL Endpoint (Sorgulama Noktası) olarak yayımlanan RDF dokümanları, farklı servislerin verileri ile aynı anda sorgulanabilir duruma getirilmiştir. Semantik Web’de yayımlanan verileri kullanarak uygulama geliştirmek isteyenlere, SPARQL sorgulamaları çok büyük veri setleri içinden seçtiklerine erişme esnekliği tanınır. Tez kapsamında üretilen idari birimlere ilişkin RDF dokümanları da SPARQL Endpoint ile Semantik Web uygulaması geliştirmek isteyenlerin kullanımına açılmıştır. Bağlı Veri ve SPARQL Endpoint olarak yayımlanan verilerin keşfedilebilirliğinin artması için bir Bağlı Veri Web sitesi kurulması önerilir. Çalışma kapsamında oluşturulan <http://araziyonetimi.org> sitesi bu amaca hizmet etmektedir. Web sitesi, çalışma kapsamında geliştirilen ontolojiler ve RDF örneklemelerini sunmakta, SPARQL sorgulamalarına olanak vermektedir.

Çalışma, coğrafi yer adları dizinleri bağlamında, konumsal verinin Bağlı Veri yaklaşımıyla sunumuna yönelik ülkemizdeki ilk çalışmadır. Henüz gelişmekte olan Bağlı Veri yaklaşımının gelecekte büyük potansiyel taşıdığı düşünülmektedir. Bu tezin yazıldığı günlerde, ARE3NA projesi ile UML sınıf diyagramları ile temsil edilen INSPIRE veri tanımlamalarının ontolojilere dönüştürülmesine ilişkin çalışmaları yapılmaktadır. Bu açıdan çalışma, ARE3NA projesinin idari birimler detayına odaklanan yerel bir uygulaması olarak değerlendirilebilir. Çalışma ayrıca, ‘Bağlı Arazi Yönetimi’ (Linked Land Administration) kavramına katkı sağlamayı hedeflemektedir [126]. Bu çalışmanın ülkemiz kapsamında diğer veri temaları ile genişletilmesi için, TUCBS Kavramsal Modeline benzer bir ontoloji modeli kurulması düşünülebilir.

KAYNAKLAR

- [1] Vögele, T., (2004). Spatial Information Retrieval with Place Names, Doktora Tezi, University of Bremen, Germany.
- [2] Janowicz, K. ve Keßler, C., (2008). "The Role of Ontology in Improving Gazetteer Interaction", International Journal of Geographical Information Science, 22(10): 1129-1157.
- [3] ISO 19112, (2005). Geographic Information - Spatial Referencing by Geographic Identifiers. ISO/TC 211 Geographic information/Geomatics.
- [4] Hill, L. L., (2006). Georeferencing: The Geographic Associations of Information, The MIT Press, Cambridge, Massachusetts, London.
- [5] López-Pellicer, F. J., Javier-Zarazaga, F., Mogollon-Diaz, A., Nogueras-Iso, J. ve Muro-Medrano, P.R., (2007). "The Gazetteer Content Model Issue: Could Spatial Data Infrastructures Provide it?", Proceeding of the 10th AGILE Conference, 8-11 May, 187-200, Aalborg, Denmark.
- [6] EDINA, (2007). European Data Models Documentation, University of Edinburgh.
- [7] INSPIRE D2.8.I.3, (2007). INSPIRE Data Specification on Geographical Names – Guidelines,
http://inspire.ec.europa.eu/documents/Data_Specifications/INSPIRE_DataSpecification_GN_v3.0.pdf, 11/11/2014.
- [8] Goodchild, M. F. ve Hill, L., (2008). "Introduction to Digital Gazetteer Research", International Journal of Geographical Information Science, 22(10): 1039-1044.
- [9] Hastings, J. T., (2008). "Automated Conflation of Digital Gazetteer Data", International Journal of Geographical Information Science, 22(10): 1109-1127.
- [10] Southall, H., Mostern, R. ve Berman, M. L., (2011). "On Historical Gazetteers", International Journal of Humanities and Arts Computing 5(2): 127-145.
- [11] Geonames (2012), GeoNames Ontology
<http://www.geonames.org/ontology/documentation.html>, 11/11/2014.

- [12] Goodwin, J., Dolbear, C. ve Hart, G., (2008). "Geographical Linked Data: The Administrative Geography of Great Britain on the Semantic Web", Transactions in GIS, 2008, 12(1): 19-30.
- [13] Berners-Lee, T., (1998), Semantic Web Road Map, <http://www.w3.org/DesignIssues/Semantic.html>, 11/11/2014.
- [14] Heath, T. ve Bizer, C., (2011) Linked Data: Evolving the Web into a Global Data Space, 1st edition, Morgan & Claypool.
- [15] Bizer, C., Heath, T. ve Berners-Lee, T., (2009). "Linked Data - The Story So Far", International Journal of Semantic Web Infrastructure Systems, 5(3): 1-22.
- [16] Sauerman L. ve Cyganiak R., (2008), Cool URIs for Semantic Web, <http://www.w3.org/TR/cooluris/>, 11/11/2014.
- [17] Hart, G. ve Dolbear, C., (2012). Linked Data: A Geographic Perspective. CRC Press, Taylor and Francis, London.
- [18] Gruber, T. R., (1993). "A Translation Approach to Portable Ontology Specifications", Knowledge Acquisition, 5(2): 199-220.
- [19] Miles, A. ve Bechhofer S., (2009), SKOS Simple Knowledge Organization System Reference, <http://www.w3.org/TR/skos-reference/>, 11/11/2014.
- [20] Brickley, D. ve Guha, R. V., (2004), RDF Schema 1.0, <http://www.w3.org/TR/2004/REC-rdf-schema-20040210/>, 11/11/2014.
- [21] Brickley, D. ve Guha, R. V., (2014), RDF Schema 1.1, <http://www.w3.org/TR/rdf-schema/>, 11/11/2014.
- [22] Smith, M. K., Welty, C. ve McGuinness, D. L., (2004), OWL Web Ontology Language Guide, <http://www.w3.org/TR/2004/REC-owl-guide-20040210/>, 11/11/2014.
- [23] Motik, B., Parsia, B. ve Patel-Schneider, P. F., (2012), OWL 2 Web Ontology Language XML Serialization (Second Edition), <http://www.w3.org/TR/2012/REC-owl2-xml-serialization-20121211/>, 11/11/2014.
- [24] Prud'hommeaux, E. ve Seaborne A., (2008), <http://www.w3.org/TR/rdf-sparql-query/>, 11/11/2014.
- [25] DuCharme, B., (2011). "Learning SPARQL", First Edition, O'Reilly.
- [26] OGC 11-052r4, (2011). OGC GeoSPARQL - A Geographic Query Language for RDF Data, Open Geospatial Consortium, Version 1.0.
- [27] Torun, A., (2002). "Türkiye Yerleşim Yerlerinin Veritabanının (TYYV) İlişkisel Model Kullanarak Tasarımı", Harita Dergisi, 128.
- [28] Tarn, W.W., (1923). "Alexander and the Ganges", The Journal of Hellenic Studies, 43(2): 93-101.
- [29] Byzantii, S., Aristotelis Ethica Nicomachea, <http://google.com.tr/books?id=sZs->

[AAAAcAAJ&printsec=frontcover&lr=&as_brr=0&source=gb_ssimilrbooks_r&edir_esc=y#v=onepage&q&f=false](#), 11 Kasim 2014.

- [30] Hargett, J.M., (1996). "Song Dynasty Local Gazetteers and Their Place in the History of Difangzhi Writing", *Harvard Journal of Asiatic Studies*, 56: 405-442.
- [31] The Columbia Electronic Encyclopedia, Gazetteer, <http://global.britannica.com/EBchecked/topic/126960/Columbia-Encyclopedia> (<http://www.gurunet.com/>), 11/11/2014.
- [32] Rumsey, D., The Gazetteer of the States of Illinois and Missouri, <http://www.davidrumsey.com/maps5295.html>, 11/11/2014.
- [33] Jones, C. B., Alani, H. ve Tudhope, D., (2001). "Geographical Information Retrieval with Ontologies of Place", *Proceedings of the International Conference on Spatial Information Theory: Foundations of Geographic Information Science (COSIT)*, 19-23 September, 322-335, Morro Bay, CA, USA.
- [34] Burrough, P.A. ve Frank, A.U., (1996). *Geographic Objects with Indeterminate Boundaries*, Taylor & Francis, London.
- [35] Keßler, C., Janowicz, K. ve Bishr, M., (2009). "An Agenda for the Next Generation Gazetteer: Geographic Information Contribution and Retrieval", In *Proceedings of the ACM International Conference on Advances in Geographic Information Systems*, 4-6 November, 91-100, Seattle, Washington, USA.
- [36] Jones, C. B., Purves, R. S., Clough, P. D. ve Joho, H., (2008). "Modelling Vague Places With Knowledge From The Web", *International Journal of Geographical Information Science*, 22(10): 1045-1065.
- [37] Montello, D., Goodchild, M. F., Gottsegen J. ve Fohl P., (2003). "Where's downtown?: behavioural methods for determining referents of vague spatial queries", *Spatial Cognition and Computation*. 3(2,3): 185-204.
- [38] Rees, R., (2003). "Clarity in The Usage of The Terms Ontology, Taxonomy and Classification", *CIB Report*, 284: 432,439.
- [39] OGC 07-036, (2002). *Geography Markup Language Implementation Specification*, Open Geospatial Consortium Inc., Version 2.1.2.
- [40] ISO 19107, (2003). *Geographic information - Spatial schema*, ISO/TC 211 Geographic information/Geomatics.
- [41] Alexandria Digital Library, ADL Bucket Report DTD, (Revision 2004/02/22). University of California at Santa Barbara. <http://www.alexandria.ucsb.edu/middleware/dtds/ADL-bucket-report.dtd>, 11/11/2014.
- [42] OGC 99-049, (1999). *OpenGIS Simple Features Specification for SQL*, Open Geospatial Consortium Inc., Version 1.0.
- [43] Janée G. ve Hill L. L., (2003). *The ADL Gazetteer Protocol*, Version 1.2, <http://legacy.alexandria.ucsb.edu/gazetteer/>, 11/11/2014.

- [44] ISO 19118, (2011). Geographic information – Encoding, ISO/TC 211 Geographic information/Geomatics.
- [45] OGC 05-035r2, (2006). Gazetteer Service - Application Profile of the Web Feature Service Implementation Specification, Open Geospatial Consortium Inc., Version 0.9.3.
- [46] OGC 11-122r1, (2012). Best Practices Document: Gazetteer Service – Application profile of the Web Feature Service Implementation Specification, Open Geospatial Consortium Inc., Version 1.0.
- [47] The Getty Thesaurus of Geographic Names, About the TGN <http://www.getty.edu/research/tools/vocabularies/tgn/about.html> 11/11/2014.
- [48] EuroGeographics, EuroGeoNames, <http://www.eurogeographics.org/eurogeonames>, 11/11/2014.
- [49] Alexandria Digital Library Project, Gazetteer Service Protocol, <http://legacy.alexandria.ucsb.edu/gazetteer/>, 11/11/2014.
- [50] OGC 04-039, (2004). Geospatial Portal Reference Architecture: A community guide to implementing standards-based geospatial portals , Open Geospatial Consortium Inc., Version 0.2.
- [51] OGC 07-057r7, (2010). OpenGIS Web Map Tile Service Implementation Standard, Open Geospatial Consortium Inc., Version 1.0.
- [52] EDINA, (2008). European Data Models Documentation UML representation of the Conceptual schema & documentation, University of Edinburgh.
- [53] Berners-Lee, T., Hendler, J. ve Lassila, O., (2001). "The Semantic Web", Scientific American, 29-37.
- [54] Berners-Lee, T., (2006), Linked Data - Design Issues, <http://www.w3.org/DesignIssues/LinkedData.html>, 11/11/2014.
- [55] Hawke, S., (2010), An Introduction to Linked Data with Sandro Hawke, <http://vimeo.com/12444260>, 11/11/2014.
- [56] World Wide Web Consortium, W3C Semantic Web Activity, <http://www.w3.org/2007/03/layerCake-small.png>, 11/11/2014.
- [57] Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R. ve Ives, Z., (2007). "Dbpedia: A nucleus for a web of open data", The Semantic Web, Springer Berlin, 722-735.
- [58] Bizer, C., Lehmann, J., Kobilarov, G., Auer, S., Becker, C., Cyganiak, R. ve Hellmann, S., (2009). "DBpedia-A crystallization point for the Web of Data", Web Semantics, 7(3):154-165.
- [59] University of Mannheim, The Linking Open Data Cloud Diagram, <http://lod-cloud.net/>, 11/11/2014.
- [60] United Kingdom Location, (2010), A Guide to the Benefits of Linked Data and the UK Location Strategy, England.

- [61] Berners-Lee, T., Fielding, R., Irvine, U. C. ve Masinter L., (1998), Uniform Resource Identifiers (URI): Generic Syntax, <http://www.ietf.org/rfc/rfc2396.txt>, 11/11/2014.
- [62] Walsh, N. ve Williams, S., (2002), What Does a URI Identify?, <http://www.w3.org/2001/tag/doc/identify>, 11/11/2014.
- [63] Berners-Lee, T., (2002), What do HTTP URIs Identify?, <http://www.w3.org/DesignIssues/HTTP-URI.html>, 11/11/2014.
- [64] Wikipedia, Uniform Resource Name, http://en.wikipedia.org/wiki/Uniform_resource_name, 11/11/2014.
- [65] Duerst, M. ve Suignard, M., (2005), Internationalized Resource Identifiers (IRIs), <http://www.ietf.org/rfc/rfc3987.txt>, 11/11/2014.
- [66] Twitter, URL, URI, IRI, <https://twitter.com/kidehen/status/510449181748908032>, 11/11/2014.
- [67] Thompson, S. H. ve Orchard, D., (2006), URNs, Namespaces and Registries, <http://www.w3.org/2001/tag/doc/URNsAndRegistries-50>, 11/11/2014.
- [68] Bray, T., Hollander, D., Layman, A. ve Tobin, R., (2006), Namespaces in XML 1.1 (Second Edition), <http://www.w3.org/TR/xml-names11/>, 11/11/2014.
- [69] Booth, D., (2003), Four Uses of a URL: Name, Concept, Web Location and Document Instance, http://www.w3.org/2002/11/dbooth-names/dbooth-names_clean.htm, 11/11/2014.
- [70] World Wide Web Consortium, Cool URIs don't Change, <http://www.w3.org/Provider/Style/URI.html>, 11/11/2014.
- [71] Berners-Lee, T., (1997), URI References: Fragment Identifiers on URIs, <http://www.w3.org/DesignIssues/Fragment.html>, 11/11/2014.
- [72] INSPIRE D2.5, (2007). Drafting Team "Data Specifications" Generic Conceptual Model, http://inspire.ec.europa.eu/reports/ImplementingRules/inspireDataspecD2_5_v2.0.pdf, 11/11/2014.
- [73] Chief Technology Officer Council, Designing URI Sets for the UK Public Sector, https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/60975/designing-URI-sets-uk-public-sector.pdf, 11/11/2014.
- [74] Archer, P., Goedertier, S. ve Loutas, N., (2012), Study on persistent URIs, with identification of best practices and recommendations on the topic for the MSs and the EC, <https://joinup.ec.europa.eu/community/semic/document/10-rules-persistent-uris>, 11/11/2014.
- [75] Cyganiak, R., Wood, D. ve Lanthaler, M., (2014), RDF 1.1 Concepts and Abstract Syntax, <http://www.w3.org/TR/2014/REC-rdf11-concepts-20140225/>, 11/11/2014.
- [76] Manola, P. ve Miller, E., (2004), RDF Primer, <http://www.w3.org/TR/2004/REC-rdf-primer-20040210/>, 11/11/2014.

- [77] Berners-Lee, T., (1998), Why RDF model is different from the XML model, <http://www.w3.org/DesignIssues/RDF-XML>, 11/11/2014.
- [78] Gandon, F. ve Schreiber, G., (2014), RDF 1.1 XML Syntax, <http://www.w3.org/TR/2014/REC-rdf-syntax-grammar-20140225/>, 11/11/2014.
- [79] Hayes, P., (2014), RDF Semantics, <http://www.w3.org/TR/2004/REC-rdf-mt-20040210/>, 11/11/2014.
- [80] Wood, D., (2014), What's New in RDF 1.1, <http://www.w3.org/TR/2014/NOTE-rdf11-new-20140225/>, 11/11/2014.
- [81] Herman, I., Adida, B., Sporny, M. ve Birbeck, M., (2013), RDFa 1.1 Primer - Second Edition Rich Structured Data Markup for Web Documents, <http://www.w3.org/TR/rdfa-primer/>, 11/11/2014.
- [82] Prud'hommeaux, E. ve Carothers, G., (2014), RDF 1.1 Turtle Terse RDF Triple Language, <http://www.w3.org/TR/2014/REC-turtle-20140225/>, 11/11/2014.
- [83] Carothers, G. ve Seaborne, A., (2014), RDF 1.1 N-Triples A line-based syntax for an RDF graph, <http://www.w3.org/TR/2014/REC-n-triples-20140225/>, 11/11/2014.
- [84] Sporny, M., Kellogg, G. ve Lanthaler, M., (2014), JSON-LD 1.0 A JSON-based Serialization for Linked Data, <http://www.w3.org/TR/json-ld/>, 11/11/2014.
- [85] OpenLink, Virtuoso Programmer's Guide - RDF Middleware ("Sponger"), <http://virtuoso.openlinksw.com/whitepapers/sponger%20cartridge%20programmers%20guide%20rdf%20middleware.html>, 11/11/2014.
- [86] OpenLink, Sponger Cartridges (RDF Mappers or RDFizers) for Virtuoso, <http://virtuoso.openlinksw.com/dataspace/doc/dav/wiki/Main/VirtProgrammerGuideRDFCartridge>, 11/11/2014.
- [87] Open Semantic Framework, RDFizers, <http://wiki.opensemanticframework.org/index.php/RDFizers>, 11/11/2014.
- [88] Bulen, A., Carter, J. J., Varanka, D. E., (2011), A Program for the Conversion of The National Map Data from Proprietary Format to Resource Description Framework (RDF), <http://pubs.usgs.gov/of/2011/1142/pdf/OFR2011-1142.pdf>, 11/11/2014.
- [89] European Commission ISA Joinup, A Reusable INSPIRE Reference Platform, <https://joinup.ec.europa.eu/community/are3na/description>, 11/11/2014.
- [90] Mark, D., Smith, B. ve Tversky, B., (1999). "Ontology and Geographic Objects: An Empirical Study of Cognitive Categorization", Lecture Notes in Computer Science, 1661: 283–298.
- [91] Torres, M., Quintero, R., Moreno-Ibarra, M., Menchaca-Mendez, R. ve Guzman, G., (2011). "GEONTO-MET: An Approach to Conceptualizing the

- Geographic Domain”, International Journal of Geographical Information Science, 25(10): 1633-1657.
- [92] Çağdaş, V., ve Stubkjær, E., (2012), Cadastre and Land Administration Thesaurus (CaLAThe), <http://cadastralvocabulary.org/>, 11/11/2014.
 - [93] Horridge, M. ve Patel-Schneider, P. F., (2012), OWL 2 Web Ontology Language Manchester Syntax (Second Edition), <http://www.w3.org/TR/owl2-manchester-syntax/>, 11/11/2014.
 - [94] Horridge, M., Drummond, N., Goodwin, J., Rector, A., Stevens, R. ve Wang, H. H., (2006). “The Manchester OWL Syntax”, In OWLED.
 - [95] Patel-Schneider, P. F., Hayes, P. ve Horrocks, I., (2004), OWL Web Ontology Language Semantics and Abstract Syntax, <http://www.w3.org/TR/2004/REC-owl-semantics-20040210/>, 11/11/2014.
 - [96] Carroll, J., Herman, I. ve Patel-Schneider, P. F., (2012), OWL 2 Web Ontology Language RDF-Based Semantics (Second Edition), <http://www.w3.org/TR/2012/REC-owl2-rdf-based-semantics-20121211/>, 11/11/2014.
 - [97] Noy, N. F. ve McGuinness, D. L., (2001). “Ontology Development 101: a Guide to Creating Your First Ontology”, Stanford Knowledge Systems Laboratory Technical Report KSL-01-05, Stanford Medical Informatics Technical Report SMI-2001-0880.
 - [98] Rector, A. L., Drummond, N., Horridge, M., Rogers, J., Knublauch, H., Stevens, R., Wang, H. ve Wroe, C., (2004). “OWL pizzas: Practical Experience of Teaching OWL-DL: Common Errors & Common Patterns”, In Proc. of EKAW, 62-81.
 - [99] Bechhofer, S., Harmelen, F., Hendler, J., Horrocks, I., McGuinness, D. L., Patel-Schneider, P. F. ve Stein, L. A., (2004), OWL Web Ontology Language Reference, <http://www.w3.org/TR/owl-ref/>, 11/11/2014.
 - [100] Motik, B., Patel-Schneider, P. F. ve Parsia, B., (2012), OWL 2 Web Ontology Language Structural Specification and Functional-Style Syntax (Second Edition), <http://www.w3.org/TR/2012/REC-owl2-syntax-20121211/>, 11/11/2014.
 - [101] Fernandez, M., Gomez-Perez, A. ve Juristo, N., (1997). “METHONTOLOGY: From Ontological Art Towards Ontological Engineering”, Proceedings of the Ontological Engineering AAAI-97 Spring Symposium Series, 24-26 March.
 - [102] DERI, Neologism, <http://neologism.deri.ie/>, 11/11/2014.
 - [103] NeOn, The NeOn toolkit, http://neon-toolkit.org/wiki/Main_Page, 11/11/2014.
 - [104] Revelytix, Knoodl, <http://www.revelytix.com/?q=taxonomy/term/4>, 11/11/2014.
 - [105] Apache Jena, ARQ - A SPARQL Processor for Jena, <http://jena.apache.org/documentation/query/>, 11/11/2014.

- [106] Apache Jena, Fuseki: serving RDF data over HTTP, http://jena.apache.org/documentation/serving_data/, 11/11/2014.
- [107] Kolas, D., (2008). "Supporting Spatial Semantics with SPARQL", Transactions in GIS, 12(1): 5-18.
- [108] Salas, J. M. ve Harth, A., (2011). "Finding Spatial Equivalences Across Multiple RDF Datasets", Terra Cognita 2011 Workshop Foundations, Technologies and Applications of the Geospatial Web, 114-126.
- [109] Volz, J., Bizer, C., Gaedke, M. ve Kobilarov, G., (2009). "Discovering and Maintaining Links on the Web of Data", The Semantic Web-ISWC 2009, 650-665.
- [110] Berners-Lee, T., (1998), What the Semantic Web can represent, <http://www.w3.org/DesignIssues/RDFnot.html>, 11/11/2014.
- [111] Berrueta, D. ve Phipps, J., (1998), Best Practice Recipes for Publishing RDF Vocabularies, <http://www.w3.org/TR/swbp-vocab-pub/>, 11/11/2014.
- [112] Das, S., Sundara, S. ve Cyganiak, R., (2012), R2RML: RDB to RDF Mapping Language, <http://www.w3.org/TR/r2rml/>, 11/11/2014.
- [113] Hert, M., Reif, G. ve Gall, H. C., (2011). "A Comparison of RDB-to-RDF Mapping Languages", Proceeding Semantics '11 Proceedings of the 7th International Conference on Semantic Systems, USA, 25-32.
- [114] Auer, S., Dietzold, S., Lehmann, J., Hellmann, S. ve Aumüller, D., (2009). "Triplify: light-weight linked data publication from relational databases", Proceedings of the 18th International Conference on World Wide Web, 621-630.
- [115] Stadler, C., Lehmann, J., Höffner, K. ve Auer, S., (2012). "LinkedGeoData: A Core for a Web of Spatial Open Data", Semantic Web 3(4): 333-354.
- [116] Doğanay, H., (1997). "Türkiye Beşeri Coğrafyası", Millî Eğitim Basımevi, İstanbul.
- [117] Ortaylı, İ., (1999). "Tanzimat Devri ve Sonrası İdârî Teşkilât", Osmanlı Devleti Tarihi (Osmanlı Devleti'nin Kuruluşu'nun 700. Yılı Armağanı), İstanbul.
- [118] Çadırcı, M., (2007). "Tanzimat Sürecinde Türkiye - Ülke Yönetimi", İmge, Ankara.
- [119] Ortaylı, İ., (1976). "Türkiye'de Taşra Yönetim ve Yöneticiliğinin Evrimi", TİD Bilimsel Araştırma Dizisi 1, Ankara.
- [120] Sayan, İ. Ö., (2013), Türkiye'de İdari Sistem Ve Örgütlenme, http://www.kas.de/wf/doc/kas_34517-1522-12-30.pdf?130529092810, 11/11/2014.
- [121] Batuk F., Öztürk D. ve Enem O., (2007). "Türkiye Ulusal Konumsal Veri Altyapısı İçin Temel Veriler", HKM, 96.
- [122] Küçükyağcı N., (2013) "Bucakların İşlevsizleştirilmesinde Merkezin/Merkezi Siyasetin Rolü", Türkiye Cumhuriyeti Başbakanlık, Ankara.

- [123] CBS Genel Müdürlüğü, “TUCBS.ID İdari Birim Veri Teması”, Çevre ve Şehircilik Bakanlığı, TUCBS ID-001, V1.1, TUCBS Uygulama Esasları, (Editör: A.Ç.Aydinoğlu, B.Uzun), Aralık, 2012.
- [124] CBS Genel Müdürlüğü, “ TUCBS.AD Adres Veri Teması”, Çevre ve Şehircilik Bakanlığı, TUCBS AD-001, V1.1, TUCBS Uygulama Esasları, (Editör: A.Ç.Aydinoğlu, A. Kara, T.Yomralıoğlu), Aralık, 2012.
- [125] Lopez, F.J., Vilches, L.M., Zarazaga, F.J., Muro, P.R., Corcho, O., “The Delft Report:Linked Data and the challenges for geographic information standardization”, Revista Catalana de Geografia, 17(44), 2012.
- [126] Çağdaş V., Stubkjær, E., in press. Core immovable property vocabulary for European linked land administration. Survey Review.

GEOMETRİ ONTOLOJİSİ

```
<?xml version="1.0"?>
```

```
<!DOCTYPE rdf:RDF [  
  <!ENTITY owl "http://www.w3.org/2002/07/owl#" >  
  <!ENTITY dc "http://purl.org/dc/elements/1.1/" >  
  <!ENTITY xsd "http://www.w3.org/2001/XMLSchema#" >  
  <!ENTITY skos "http://www.w3.org/2004/02/skos/core#" >  
  <!ENTITY rdfs "http://www.w3.org/2000/01/rdf-schema#" >  
  <!ENTITY rdf "http://www.w3.org/1999/02/22-rdf-syntax-ns#" >  
>
```

```
<rdf:RDF xmlns="http://araziyonetimi.org/ontoloji/geometri/"  
  xml:base="http://araziyonetimi.org/ontoloji/geometri/"  
  xmlns:dc="http://purl.org/dc/elements/1.1/"  
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"  
  xmlns:owl="http://www.w3.org/2002/07/owl#"  
  xmlns:xsd="http://www.w3.org/2001/XMLSchema#"  
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"  
  xmlns:skos="http://www.w3.org/2004/02/skos/core#">  
  <owl:Ontology rdf:about="http://araziyonetimi.org/ontoloji/geometri/">  
    <rdfs:label xml:lang="tr">Geometri Ontolojisi</rdfs:label>  
    <dc:date rdf:datatype="&xsd;dateTime">01.12.2014</dc:date>  
    <owl:versionInfo rdf:datatype="&xsd;decimal">1.0</owl:versionInfo>  
    <skos:editorialNote rdf:datatype="&xsd;string">Geometri Ontolojisinin, GeoSPARQL ile uyumlu olarak 01.12.2014 tarihinde yayımlanan, 1.0 versiyonudur.</skos:editorialNote>  
    <rdf:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/geometri.owl"/>  
    <owl:imports rdf:resource="http://www.opengis.net/ont/geosparql"/>  
  </owl:Ontology>
```

```

<!--
////////////////////////////////////
//
// Annotation properties
//
////////////////////////////////////
-->

```

```

<!-- http://www.w3.org/2000/01/rdf-schema#comment -->

```

```

<owl:AnnotationProperty rdf:about="&rdfs;comment"/>

```

```

<!-- http://www.w3.org/2000/01/rdf-schema#isDefinedBy -->

```

```

<owl:AnnotationProperty rdf:about="&rdfs;isDefinedBy"/>

```

```

<!-- http://www.w3.org/2000/01/rdf-schema#label -->

```

```

<owl:AnnotationProperty rdf:about="&rdfs;label"/>

```

```

<!--
////////////////////////////////////
//
// Object Properties
//
////////////////////////////////////
-->

```

```

<!-- http://araziyonetimi.org/ontoloji/geometri/koordinat -->

```

```

<owl:ObjectProperty rdf:about="http://araziyonetimi.org/ontoloji/geometri/koordinat">

```

```

<rdfs:label xml:lang="tr">koordinat</rdfs:label>
<rdfs:comment xml:lang="tr">2 boyutlu geometrileri ifade eder.</rdfs:comment>
<rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/geometri.owl"/>
<rdfs:range rdf:resource="http://www.opengis.net/ont/geosparql#Geometry"/>
<owl:equivalentProperty rdf:resource="http://www.opengis.net/ont/geosparql#hasGeometry"/>
<rdfs:domain>
  <owl:Restriction>
    <owl:onProperty rdf:resource="http://www.opengis.net/ont/geosparql#asGML"/>
    <owl:someValuesFrom rdf:resource="http://www.opengis.net/ont/geosparql#gmlLiteral"/>
  </owl:Restriction>
</rdfs:domain>
</owl:ObjectProperty>

<!-- http://www.opengis.net/ont/geosparql#hasGeometry -->

<rdf:Description rdf:about="http://www.opengis.net/ont/geosparql#hasGeometry"/>

<!--
////////////////////////////////////
//
// Data properties
//
////////////////////////////////////
-->

<!-- http://araziyonetimi.org/ontoloji/geometri/gmlOlarak -->

<owl:DatatypeProperty rdf:about="http://araziyonetimi.org/ontoloji/geometri/gmlOlarak">
  <rdfs:label xml:lang="tr">GML olarak</rdfs:label>
  <rdfs:comment xml:lang="tr">Geometri tiplerinin GML olarak sunumunu ifade eder. XMLLiteral olarak saklanır.</rdfs:comment>
  <rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/geometri.owl"/>
  <rdfs:domain rdf:resource="http://araziyonetimi.org/ontoloji/geometri/Geometri"/>
  <owl:equivalentProperty rdf:resource="http://www.opengis.net/ont/geosparql#asGML"/>
  <rdfs:range rdf:resource="&rdf;XMLLiteral"/>
</owl:DatatypeProperty>

```

```

<!-- http://www.opengis.net/ont/geosparql#asGML -->

<rdf:Description rdf:about="http://www.opengis.net/ont/geosparql#asGML"/>


<!--
////////////////////////////////////
//
// Classes
//
////////////////////////////////////
-->


<!-- http://araziyonetimi.org/ontoloji/geometri/Geometri -->

<owl:Class rdf:about="http://araziyonetimi.org/ontoloji/geometri/Geometri">
  <rdfs:label xml:lang="tr">Geometri</rdfs:label>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/geometri/gmlOlarak"/>
      <owl:minQualifiedCardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:minQualifiedCardinality>
      <owl:onDataRange rdf:resource="&rdf;XMLLiteral"/>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:comment xml:lang="tr">Nokta, çizgi ve poligon geometrilerinin süper sınıfıdır. </rdfs:comment>
  <rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/geometri.owl"/>
</owl:Class>
</rdf:RDF>


<!-- Generated by the OWL API (version 3.4.2) http://owlapi.sourceforge.net -->

```

KONUMSAL İLİŞKİLER ONTOLOJİSİ

```
<?xml version="1.0"?>
```

```
<!DOCTYPE rdf:RDF [
```

```
  <!ENTITY dcterm "http://purl.org/dc/terms/" >
  <!ENTITY owl "http://www.w3.org/2002/07/owl#" >
  <!ENTITY dc "http://purl.org/dc/elements/1.1/" >
  <!ENTITY xsd "http://www.w3.org/2001/XMLSchema#" >
  <!ENTITY skos "http://www.w3.org/2004/02/skos/core#" >
  <!ENTITY rdfs "http://www.w3.org/2000/01/rdf-schema#" >
  <!ENTITY rdf "http://www.w3.org/1999/02/22-rdf-syntax-ns#" >
```



```
<rdf:RDF xmlns="http://araziyonetimi.org/ontoloji/konumsaliliskiler/"
```

```
  xml:base="http://araziyonetimi.org/ontoloji/konumsaliliskiler/"
```

```
  xmlns:dc="http://purl.org/dc/elements/1.1/"
```

```
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
```

```
  xmlns:owl="http://www.w3.org/2002/07/owl#"
```

```
  xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
```

```
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
```

```
  xmlns:skos="http://www.w3.org/2004/02/skos/core#"
```

```
  xmlns:dcterm="http://purl.org/dc/terms/">
```

```
<owl:Ontology rdf:about="http://araziyonetimi.org/ontoloji/konumsaliliskiler/">
```

```
  <rdfs:label xml:lang="tr">Konumsal İlişkiler Ontolojisi</rdfs:label>
```

```
  <dc:date rdf:datatype="&xsd:date">01.12.2014</dc:date>
```

```
  <owl:versionInfo rdf:datatype="&xsd:string">1.0</owl:versionInfo>
```

```
  <skos:editorialNote rdf:datatype="&xsd:string">Konumsal ilişkiler ontolojisinin; içerir, içindedir, dokunur ve eşittir ilişkilerini  
içeren; 01.12.2014 tarihli 1.0 versiyonudur. Konumsal ilişkiler GeoSPARQL konumsal ilişkileri ile eşitlenmiştir.</skos:editorialNote>
```

```
  <dc:contributor xml:lang="tr">Abdullah Kara, Volkan Çağdaş</dc:contributor>
```

```

<dc:title xml:lang="tr">Konumsal İlişkiler Ontolojisi</dc:title>
<dc:description xml:lang="tr">Konumsal ilişkileri ifade eden ontolojidir.</dc:description>
<rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/konumsaliliskiler.owl"/>
<owl:imports rdf:resource="http://www.opengis.net/ont/geosparql"/>
</owl:Ontology>

```

```

<!--
////////////////////////////////////
//
// Annotation properties
//
////////////////////////////////////
-->

```

```

<!-- http://purl.org/dc/elements/1.1/contributor -->

```

```

<owl:AnnotationProperty rdf:about="&dc;contributor"/>

```

```

<!-- http://purl.org/dc/elements/1.1/description -->

```

```

<owl:AnnotationProperty rdf:about="&dc;description"/>

```

```

<!-- http://purl.org/dc/elements/1.1/title -->

```

```

<owl:AnnotationProperty rdf:about="&dc;title"/>

```

```

<!-- http://www.w3.org/2000/01/rdf-schema#comment -->

```

```

<owl:AnnotationProperty rdf:about="&rdfs;comment"/>

```

```

<!-- http://www.w3.org/2000/01/rdf-schema#isDefinedBy -->

```

```
<owl:AnnotationProperty rdf:about="&rdfs;isDefinedBy"/>
```

```
<!-- http://www.w3.org/2000/01/rdf-schema#label -->
```

```
<owl:AnnotationProperty rdf:about="&rdfs;label"/>
```

```
<!--  
/////////////////////////////////////  
//  
// Object Properties  
//  
/////////////////////////////////////  
-->
```

```
<!-- http://araziyonetimi.org/ontoloji/konumsaliliskiler/dokunur -->
```

```
<owl:ObjectProperty rdf:about="http://araziyonetimi.org/ontoloji/konumsaliliskiler/dokunur">  
  <rdf:type rdf:resource="&owl;SymmetricProperty"/>  
  <rdfs:label xml:lang="tr">dokunur</rdfs:label>  
  <rdfs:comment xml:lang="tr">iki nesnenin sınırlarının keşisip, iç bölgelerinin ayrı olması durumudur.</rdfs:comment>  
  <rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/konumsaliliskiler.owl"/>  
  <rdfs:domain rdf:resource="http://www.opengis.net/ont/geosparql#SpatialObject"/>  
  <rdfs:range rdf:resource="http://www.opengis.net/ont/geosparql#SpatialObject"/>  
  <owl:equivalentProperty rdf:resource="http://www.opengis.net/ont/geosparql#sfTouches"/>  
  <owl:propertyChainAxiom rdf:parseType="Collection">  
    <rdf:Description rdf:about="http://araziyonetimi.org/ontoloji/konumsaliliskiler/esittir"/>  
    <rdf:Description rdf:about="http://araziyonetimi.org/ontoloji/konumsaliliskiler/dokunur"/>  
  </owl:propertyChainAxiom>  
  <owl:propertyChainAxiom rdf:parseType="Collection">  
    <rdf:Description rdf:about="http://araziyonetimi.org/ontoloji/konumsaliliskiler/dokunur"/>  
    <rdf:Description rdf:about="http://araziyonetimi.org/ontoloji/konumsaliliskiler/esittir"/>  
  </owl:propertyChainAxiom>  
</owl:ObjectProperty>
```

<!-- http://araziyonetimi.org/ontoloji/konumsaliliskiler/esittir -->

```
<owl:ObjectProperty rdf:about="http://araziyonetimi.org/ontoloji/konumsaliliskiler/esittir">
  <rdf:type rdf:resource="&owl;ReflexiveProperty"/>
  <rdf:type rdf:resource="&owl;SymmetricProperty"/>
  <rdf:type rdf:resource="&owl;TransitiveProperty"/>
  <rdfs:label xml:lang="tr">eşittir</rdfs:label>
  <rdfs:comment xml:lang="tr">İki nesnenin aynı sınıra ve aynı iç bölgeye ait olması durumudur.</rdfs:comment>
  <rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/konumsaliliskiler.owl"/>
  <rdfs:range rdf:resource="http://www.opengis.net/ont/geosparql#SpatialObject"/>
  <rdfs:domain rdf:resource="http://www.opengis.net/ont/geosparql#SpatialObject"/>
  <owl:equivalentProperty rdf:resource="http://www.opengis.net/ont/geosparql#sfEquals"/>
</owl:ObjectProperty>
```

<!-- http://araziyonetimi.org/ontoloji/konumsaliliskiler/icerir -->

```
<owl:ObjectProperty rdf:about="http://araziyonetimi.org/ontoloji/konumsaliliskiler/icerir">
  <rdf:type rdf:resource="&owl;TransitiveProperty"/>
  <rdfs:label xml:lang="tr">içerir</rdfs:label>
  <rdfs:comment xml:lang="tr">Bir nesnenin, bir diğerinin tamamen içirme durumudur.</rdfs:comment>
  <rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/konumsaliliskiler.owl"/>
  <owl:inverseOf rdf:resource="http://araziyonetimi.org/ontoloji/konumsaliliskiler/icindedir"/>
  <rdfs:domain rdf:resource="http://www.opengis.net/ont/geosparql#SpatialObject"/>
  <rdfs:range rdf:resource="http://www.opengis.net/ont/geosparql#SpatialObject"/>
  <owl:equivalentProperty rdf:resource="http://www.opengis.net/ont/geosparql#sfContains"/>
  <owl:propertyChainAxiom rdf:parseType="Collection">
    <rdf:Description rdf:about="http://araziyonetimi.org/ontoloji/konumsaliliskiler/esittir"/>
    <rdf:Description rdf:about="http://araziyonetimi.org/ontoloji/konumsaliliskiler/icerir"/>
  </owl:propertyChainAxiom>
  <owl:propertyChainAxiom rdf:parseType="Collection">
    <rdf:Description rdf:about="http://araziyonetimi.org/ontoloji/konumsaliliskiler/icerir"/>
    <rdf:Description rdf:about="http://araziyonetimi.org/ontoloji/konumsaliliskiler/esittir"/>
  </owl:propertyChainAxiom>
</owl:ObjectProperty>
```

<!-- http://araziyonetimi.org/ontoloji/konumsaliliskiler/icindedir -->

```
<owl:ObjectProperty rdf:about="http://araziyonetimi.org/ontoloji/konumsaliliskiler/icindedir">
  <rdf:type rdf:resource="&owl;TransitiveProperty"/>
  <rdfs:label xml:lang="tr">içindedir</rdfs:label>
```

```

<rdfs:comment xml:lang="tr">Bir nesnenin, bir diğèrinin tamamen ierisinde olma durumudur.</rdfs:comment>
<rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/konumsaliliskiler.owl"/>
<rdfs:range rdf:resource="http://www.opengis.net/ont/geosparql#SpatialObject"/>
<rdfs:domain rdf:resource="http://www.opengis.net/ont/geosparql#SpatialObject"/>
<owl:equivalentProperty rdf:resource="http://www.opengis.net/ont/geosparql#sfWithin"/>
<owl:propertyChainAxiom rdf:parseType="Collection">
  <rdf:Description rdf:about="http://araziyonetimi.org/ontoloji/konumsaliliskiler/icindedir"/>
  <rdf:Description rdf:about="http://araziyonetimi.org/ontoloji/konumsaliliskiler/esittir"/>
</owl:propertyChainAxiom>
<owl:propertyChainAxiom rdf:parseType="Collection">
  <rdf:Description rdf:about="http://araziyonetimi.org/ontoloji/konumsaliliskiler/esittir"/>
  <rdf:Description rdf:about="http://araziyonetimi.org/ontoloji/konumsaliliskiler/icindedir"/>
</owl:propertyChainAxiom>
</owl:ObjectProperty>

```

```

<!-- http://www.opengis.net/ont/geosparql#sfContains -->

```

```

<rdf:Description rdf:about="http://www.opengis.net/ont/geosparql#sfContains"/>

```

```

<!-- http://www.opengis.net/ont/geosparql#sfEquals -->

```

```

<rdf:Description rdf:about="http://www.opengis.net/ont/geosparql#sfEquals"/>

```

```

<!-- http://www.opengis.net/ont/geosparql#sfTouches -->

```

```

<rdf:Description rdf:about="http://www.opengis.net/ont/geosparql#sfTouches"/>

```

```

<!-- http://www.opengis.net/ont/geosparql#sfWithin -->

```

```

<rdf:Description rdf:about="http://www.opengis.net/ont/geosparql#sfWithin"/>

```

```

<!--
////////////////////////////////////
//

```

```

// Classes
//
////////////////////////////////////
-->

<!-- http://www.w3.org/2002/07/owl#Thing -->

<owl:Class rdf:about="&owl;Thing"/>

<!--
////////////////////////////////////
//
// Individuals
//
////////////////////////////////////
-->

<!-- http://araziyonetimi.org/ontoloji/konumsaliliskiler/ -->

<owl:Thing rdf:about="http://araziyonetimi.org/ontoloji/konumsaliliskiler/">
  <rdf:type rdf:resource="&owl;NamedIndividual"/>
</owl:Thing>
</rdf:RDF>

<!-- Generated by the OWL API (version 3.4.2) http://owlapi.sourceforge.net -->

```

İDARİ BİRİMLER ONTOLOJİSİ

```
<?xml version="1.0"?>
```

```
<!DOCTYPE rdf:RDF [
```

```
  <!ENTITY terms "http://purl.org/dc/terms/" >
  <!ENTITY owl "http://www.w3.org/2002/07/owl#" >
  <!ENTITY dc "http://purl.org/dc/elements/1.1/" >
  <!ENTITY xsd "http://www.w3.org/2001/XMLSchema#" >
  <!ENTITY skos "http://www.w3.org/2004/02/skos/core#" >
  <!ENTITY rdfs "http://www.w3.org/2000/01/rdf-schema#" >
  <!ENTITY rdf "http://www.w3.org/1999/02/22-rdf-syntax-ns#" >
```



```
<rdf:RDF xmlns="http://araziyonetimi.org/ontoloji/idaribirim/"
```

```
  xml:base="http://araziyonetimi.org/ontoloji/idaribirim/"
```

```
  xmlns:dc="http://purl.org/dc/elements/1.1/"
```

```
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
```

```
  xmlns:terms="http://purl.org/dc/terms/"
```

```
  xmlns:owl="http://www.w3.org/2002/07/owl#"
```

```
  xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
```

```
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
```

```
  xmlns:skos="http://www.w3.org/2004/02/skos/core#">
```

```
<owl:Ontology rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/">
```

```
  <rdfs:label xml:lang="tr">Türkiye İdari Birimler Ontolojisi</rdfs:label>
```

```
  <dc:date rdf:datatype="&xsd:date">01.12.2014</dc:date>
```

```
  <owl:versionInfo rdf:datatype="&xsd:decimal">1.0</owl:versionInfo>
```

```
  <skos:editorialNote rdf:datatype="&xsd:string">Türkiye İdari Birimler Ontolojisinin 01.12.2014 tarihinde düzenlenen 1.0 versiyonudur.</skos:editorialNote>
```

```
  <dc:creator xml:lang="tr">Abdullah Kara, Volkan Çağdaş</dc:creator>
```

```

<dc:contributor xml:lang="tr">Abdullah Kara, Volkan Çağdaş</dc:contributor>
<dc:title xml:lang="tr">Türkiye İdari Birimler Ontolojisi</dc:title>
<rdfs:comment xml:lang="tr">Türkiye&apos;de bulunan İdari Birimleri ifade eden ontolojidir.</rdfs:comment>
<owl:imports rdf:resource="http://araziyonetimi.org/ontoloji/geometri/" />
<rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim.owl" />
<owl:imports rdf:resource="http://araziyonetimi.org/ontoloji/geometri/" />
<owl:imports rdf:resource="http://araziyonetimi.org/ontoloji/konumsaliliskiler/" />
</owl:Ontology>

```

```

<!--
////////////////////////////////////
//
// Annotation properties
//
////////////////////////////////////
-->

```

```

<!-- http://purl.org/dc/elements/1.1/contributor -->

```

```

<owl:AnnotationProperty rdf:about="&dc;contributor"/>

```

```

<!-- http://purl.org/dc/elements/1.1/creator -->

```

```

<owl:AnnotationProperty rdf:about="&dc;creator"/>

```

```

<!-- http://purl.org/dc/elements/1.1/description -->

```

```

<owl:AnnotationProperty rdf:about="&dc;description"/>

```

```

<!-- http://purl.org/dc/elements/1.1/title -->

```

```

<owl:AnnotationProperty rdf:about="&dc;title"/>

```

```
<!-- http://www.w3.org/2000/01/rdf-schema#comment -->
```

```
<owl:AnnotationProperty rdf:about="&rdfs;comment"/>
```

```
<!-- http://www.w3.org/2000/01/rdf-schema#isDefinedBy -->
```

```
<owl:AnnotationProperty rdf:about="&rdfs;isDefinedBy"/>
```

```
<!-- http://www.w3.org/2000/01/rdf-schema#label -->
```

```
<owl:AnnotationProperty rdf:about="&rdfs;label"/>
```

```
<!--
```

```
////////////////////////////////////
```

```
//
```

```
// Object Properties
```

```
//
```

```
////////////////////////////////////
```

```
-->
```

```
<!-- http://araziyonetimi.org/ontoloji/geometri/koordinat -->
```

```
<owl:ObjectProperty rdf:about="http://araziyonetimi.org/ontoloji/geometri/koordinat"/>
```

```
<!-- http://araziyonetimi.org/ontoloji/idaribirim/belediyelcerir -->
```

```
<owl:ObjectProperty rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/belediyelcerir">
```

```
<rdfs:label xml:lang="tr">belediye İçerir</rdfs:label>
```

```
<rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim.owl"/>
```

```
<rdfs:domain rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/IdariBirim"/>
```

```
<rdfs:range rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/IdariBirim"/>
```

```
<rdfs:subPropertyOf rdf:resource="http://araziyonetimi.org/ontoloji/konumsaliliskiler/icerir"/>
</owl:ObjectProperty>
```

```
<!-- http://araziyonetimi.org/ontoloji/idaribirim/belediyelcindedir -->
```

```
<owl:ObjectProperty rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/belediyelcindedir">
  <rdfs:label xml:lang="tr">belediye içindedir</rdfs:label>
  <rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim.owl"/>
  <rdfs:domain rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/IdariBirim"/>
  <rdfs:range rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/IdariBirim"/>
  <owl:inverseOf rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/belediyelcerir"/>
  <rdfs:subPropertyOf rdf:resource="http://araziyonetimi.org/ontoloji/konumsaliliskiler/icindedir"/>
</owl:ObjectProperty>
```

```
<!-- http://araziyonetimi.org/ontoloji/idaribirim/illcerir -->
```

```
<owl:ObjectProperty rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/illcerir">
  <rdfs:label xml:lang="tr">il içerir</rdfs:label>
  <rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim.owl"/>
  <rdfs:range rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/IdariBirim"/>
  <rdfs:domain rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/IdariBirim"/>
  <owl:inverseOf rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/illcindedir"/>
  <rdfs:subPropertyOf rdf:resource="http://araziyonetimi.org/ontoloji/konumsaliliskiler/icerir"/>
</owl:ObjectProperty>
```

```
<!-- http://araziyonetimi.org/ontoloji/idaribirim/illcindedir -->
```

```
<owl:ObjectProperty rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/illcindedir">
  <rdfs:label xml:lang="tr">il içindedir</rdfs:label>
  <rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim.owl"/>
  <rdfs:range rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/IdariBirim"/>
  <rdfs:domain rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/IdariBirim"/>
  <rdfs:subPropertyOf rdf:resource="http://araziyonetimi.org/ontoloji/konumsaliliskiler/icindedir"/>
</owl:ObjectProperty>
```

```
<!-- http://araziyonetimi.org/ontoloji/idaribirim/ilOzelldarelcerir -->
```

```

<owl:ObjectProperty rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/ilOzelIdarelcerir">
  <rdfs:label xml:lang="tr">il Özel İdare İçerir</rdfs:label>
  <rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim.owl"/>
  <rdfs:domain rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/IdariBirim"/>
  <rdfs:range rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/IdariBirim"/>
  <rdfs:subPropertyOf rdf:resource="http://araziyonetimi.org/ontoloji/konumsaliliskiler/icerir"/>
</owl:ObjectProperty>

```

```

<!-- http://araziyonetimi.org/ontoloji/idaribirim/ilOzelIdarelcindedir -->

```

```

<owl:ObjectProperty rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/ilOzelIdarelcindedir">
  <rdfs:label xml:lang="tr">il Özel İdare İçindedir</rdfs:label>
  <rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim.owl"/>
  <rdfs:domain rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/IdariBirim"/>
  <rdfs:range rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/IdariBirim"/>
  <owl:inverseOf rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/ilOzelIdarelcerir"/>
  <rdfs:subPropertyOf rdf:resource="http://araziyonetimi.org/ontoloji/konumsaliliskiler/icindedir"/>
</owl:ObjectProperty>

```

```

<!-- http://araziyonetimi.org/ontoloji/idaribirim/ilcelcerir -->

```

```

<owl:ObjectProperty rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/ilcelcerir">
  <rdfs:label xml:lang="tr">ilçe İçerir</rdfs:label>
  <rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim.owl"/>
  <rdfs:range rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/IdariBirim"/>
  <rdfs:domain rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/IdariBirim"/>
  <rdfs:subPropertyOf rdf:resource="http://araziyonetimi.org/ontoloji/konumsaliliskiler/icerir"/>
</owl:ObjectProperty>

```

```

<!-- http://araziyonetimi.org/ontoloji/idaribirim/ilcelcindedir -->

```

```

<owl:ObjectProperty rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/ilcelcindedir">
  <rdfs:label xml:lang="tr">ilçe İçindedir</rdfs:label>
  <rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim.owl"/>
  <rdfs:range rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/IdariBirim"/>
  <rdfs:domain rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/IdariBirim"/>
  <owl:inverseOf rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/ilcelcerir"/>

```

```
<rdfs:subPropertyOf rdf:resource="http://araziyonetimi.org/ontoloji/konumsaliliskiler/icindedir"/>
</owl:ObjectProperty>
```

```
<!-- http://araziyonetimi.org/ontoloji/idaribirim/koylcerir -->
```

```
<owl:ObjectProperty rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/koylcerir">
  <rdfs:label xml:lang="tr">köy İçerir</rdfs:label>
  <rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim.owl"/>
  <rdfs:range rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/IdariBirim"/>
  <rdfs:domain rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/IdariBirim"/>
  <rdfs:subPropertyOf rdf:resource="http://araziyonetimi.org/ontoloji/konumsaliliskiler/icerir"/>
</owl:ObjectProperty>
```

```
<!-- http://araziyonetimi.org/ontoloji/idaribirim/koylcindedir -->
```

```
<owl:ObjectProperty rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/koylcindedir">
  <rdfs:label xml:lang="tr">köy İçindedir</rdfs:label>
  <rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim.owl"/>
  <rdfs:range rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/IdariBirim"/>
  <rdfs:domain rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/IdariBirim"/>
  <owl:inverseOf rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/koylcerir"/>
  <rdfs:subPropertyOf rdf:resource="http://araziyonetimi.org/ontoloji/konumsaliliskiler/icindedir"/>
</owl:ObjectProperty>
```

```
<!-- http://araziyonetimi.org/ontoloji/idaribirim/mahallelcerir -->
```

```
<owl:ObjectProperty rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/mahallelcerir">
  <rdfs:label xml:lang="tr">mahalle İçerir</rdfs:label>
  <rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim.owl"/>
  <rdfs:domain rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/IdariBirim"/>
  <rdfs:range rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/IdariBirim"/>
  <rdfs:subPropertyOf rdf:resource="http://araziyonetimi.org/ontoloji/konumsaliliskiler/icerir"/>
  <owl:propertyChainAxiom rdf:parseType="Collection">
    <rdf:Description rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/ilcelcerir"/>
    <rdf:Description rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/mahallelcerir"/>
  </owl:propertyChainAxiom>
</owl:ObjectProperty>
```

```

<!-- http://araziyonetimi.org/ontoloji/idaribirim/mahallelcindedir -->

<owl:ObjectProperty rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/mahallelcindedir">
  <rdfs:label xml:lang="tr">mahalle içindedir</rdfs:label>
  <rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim.owl"/>
  <rdfs:range rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/IdariBirim"/>
  <rdfs:domain rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/IdariBirim"/>
  <owl:inverseOf rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/mahallelcerir"/>
  <rdfs:subPropertyOf rdf:resource="http://araziyonetimi.org/ontoloji/konumsaliliskiler/icindedir"/>
</owl:ObjectProperty>

```

```

<!-- http://araziyonetimi.org/ontoloji/konumsaliliskiler/esittir -->

<owl:ObjectProperty rdf:about="http://araziyonetimi.org/ontoloji/konumsaliliskiler/esittir"/>

```

```

<!-- http://araziyonetimi.org/ontoloji/konumsaliliskiler/icerir -->

<owl:ObjectProperty rdf:about="http://araziyonetimi.org/ontoloji/konumsaliliskiler/icerir"/>

```

```

<!-- http://araziyonetimi.org/ontoloji/konumsaliliskiler/icindedir -->

<owl:ObjectProperty rdf:about="http://araziyonetimi.org/ontoloji/konumsaliliskiler/icindedir"/>

```

```

<!--
////////////////////////////////////
//
// Data properties
//
////////////////////////////////////
-->

```

<!-- http://araziyonetimi.org/ontoloji/idaribirim/NUTSKodu -->

```
<owl:DatatypeProperty rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/NUTSKodu">
  <rdfs:label xml:lang="tr">NUTS Kodu</rdfs:label>
  <dc:description xml:lang="tr">İdari birimlerin NUTS kodlarını ifade eder.</dc:description>
  <rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim.owl"/>
  <rdfs:domain rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/Mulkildare"/>
  <rdfs:range rdf:resource="&xsd;nonNegativeInteger"/>
</owl:DatatypeProperty>
```

<!-- http://araziyonetimi.org/ontoloji/idaribirim/UAVTKodu -->

```
<owl:DatatypeProperty rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/UAVTKodu">
  <rdfs:label xml:lang="tr">UAVT Kodu</rdfs:label>
  <dc:description xml:lang="tr">İdari Birimlerin UAVT kodlarını ifade eder.</dc:description>
  <rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim.owl"/>
  <rdfs:domain rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/IdariBirim"/>
  <rdfs:range rdf:resource="&xsd;nonNegativeInteger"/>
</owl:DatatypeProperty>
```

<!-- http://araziyonetimi.org/ontoloji/idaribirim/adres -->

```
<owl:DatatypeProperty rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/adres">
  <rdfs:label xml:lang="tr">adres</rdfs:label>
  <rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim.owl"/>
  <rdfs:domain rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/IdariBirim"/>
  <rdfs:range rdf:resource="&xsd:string"/>
</owl:DatatypeProperty>
```

<!-- http://araziyonetimi.org/ontoloji/idaribirim/alternatifAd -->

```
<owl:DatatypeProperty rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/alternatifAd">
  <rdfs:label xml:lang="tr">alternatif Ad</rdfs:label>
  <dc:description xml:lang="tr">İdari Birimlerin laternatif adlarını ifade eder.</dc:description>
  <rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim.owl"/>
  <rdfs:domain rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/IdariBirim"/>
  <rdfs:range rdf:resource="&xsd:string"/>
</owl:DatatypeProperty>
```

<!-- http://araziyonetimi.org/ontoloji/idaribirim/harita -->

```
<owl:DatatypeProperty rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/harita">
  <rdfs:label xml:lang="tr">harita</rdfs:label>
  <dc:description xml:lang="tr">İdari birimlerin geometrilerinin URI adresini ifade eder.</dc:description>
  <rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim.owl"/>
  <rdfs:domain rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/IdariBirim"/>
  <rdfs:range rdf:resource="&xsd:anyURI"/>
</owl:DatatypeProperty>
```

<!-- http://araziyonetimi.org/ontoloji/idaribirim/kurulusTarihi -->

```
<owl:DatatypeProperty rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/kurulusTarihi">
  <rdfs:label xml:lang="tr">kuruluş Tarihi</rdfs:label>
  <dc:description xml:lang="tr">İdari birimlerin kuruluş tarihini ifade eder.</dc:description>
  <rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim.owl"/>
  <rdfs:domain rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/IdariBirim"/>
  <rdfs:range rdf:resource="&xsd:date"/>
</owl:DatatypeProperty>
```

<!-- http://araziyonetimi.org/ontoloji/idaribirim/nufus -->

```
<owl:DatatypeProperty rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/nufus">
  <rdfs:label xml:lang="tr">nüfus</rdfs:label>
  <dc:description xml:lang="tr">İdari birimlerin nüfusunu ifade eder.</dc:description>
  <rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim.owl"/>
  <rdfs:domain rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/Mulkildare"/>
  <rdfs:range rdf:resource="&xsd;nonNegativeInteger"/>
</owl:DatatypeProperty>
```

<!-- http://araziyonetimi.org/ontoloji/idaribirim/postaKodu -->

```
<owl:DatatypeProperty rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/postaKodu">
  <rdfs:label xml:lang="tr">posta Kodu</rdfs:label>
  <dc:description xml:lang="tr">Mahallelerin posta kodlarını ifade eder.</dc:description>
```

```

<rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim.owl"/>
<rdfs:domain rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/Mahalle"/>
<rdfs:range rdf:resource="&xsd;nonNegativeInteger"/>
</owl:DatatypeProperty>

<!-- http://araziyonetimi.org/ontoloji/idaribirim/sabitTanitimNumarasi -->

<owl:DatatypeProperty rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/sabitTanitimNumarasi">
  <rdfs:label xml:lang="tr">sabit Tanitim Numarasi</rdfs:label>
  <dc:description xml:lang="tr">Mahallelerin sabit tanitim numaralarını ifade eder.</dc:description>
  <rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim.owl"/>
  <rdfs:domain rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/Mahalle"/>
  <rdfs:range rdf:resource="&xsd;nonNegativeInteger"/>
</owl:DatatypeProperty>

<!-- http://araziyonetimi.org/ontoloji/idaribirim/vikipedi -->

<owl:DatatypeProperty rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/vikipedi">
  <rdfs:label xml:lang="tr">vikipedi</rdfs:label>
  <dc:description xml:lang="tr">İdari birimlerin ilgili vikipedi sayfasını ifade eder.</dc:description>
  <rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim.owl"/>
  <rdfs:domain rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/IdariBirim"/>
  <rdfs:range rdf:resource="&xsd:anyURI"/>
</owl:DatatypeProperty>

<!-- http://araziyonetimi.org/ontoloji/idaribirim/yuzolcum -->

<owl:DatatypeProperty rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/yuzolcum">
  <rdfs:label xml:lang="tr">yüzölçüm</rdfs:label>
  <dc:description xml:lang="tr">İdari birimlerin yüzölçümlerini ifade eder.</dc:description>
  <rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim.owl"/>
  <rdfs:domain rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/IdariBirim"/>
  <rdfs:range rdf:resource="&xsd;decimal"/>
</owl:DatatypeProperty>

<!--

```

```

////////////////////////////////////
//
// Classes
//
////////////////////////////////////
-->

```

```

<!-- http://araziyonetimi.org/ontoloji/geometri/Geometri -->

```

```

<owl:Class rdf:about="http://araziyonetimi.org/ontoloji/geometri/Geometri"/>

```

```

<!-- http://araziyonetimi.org/ontoloji/idaribirim/BeldeBelediyesi -->

```

```

<owl:Class rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/BeldeBelediyesi">

```

```

  <rdfs:label xml:lang="tr">Belde Belediyesi</rdfs:label>

```

```

  <rdfs:subClassOf rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/Belediye"/>

```

```

  <rdfs:subClassOf>

```

```

    <owl:Restriction>

```

```

      <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/illcindedir"/>

```

```

      <owl:onClass rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/BuyuksehirOlmayanIl"/>

```

```

      <owl:qualifiedCardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:qualifiedCardinality>

```

```

    </owl:Restriction>

```

```

  </rdfs:subClassOf>

```

```

  <rdfs:subClassOf>

```

```

    <owl:Restriction>

```

```

      <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/koylcerir"/>

```

```

      <owl:allValuesFrom rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/Koy"/>

```

```

    </owl:Restriction>

```

```

  </rdfs:subClassOf>

```

```

  <rdfs:subClassOf>

```

```

    <owl:Restriction>

```

```

      <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/ilcelcindedir"/>

```

```

      <owl:onClass rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/BuyuksehirDisilce"/>

```

```

      <owl:qualifiedCardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:qualifiedCardinality>

```

```

    </owl:Restriction>

```

```

  </rdfs:subClassOf>

```

```

  <rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim.owl"/>

```

```

</owl:Class>

```

<!-- http://araziyonetimi.org/ontoloji/idaribirim/Belediye -->

```
<owl:Class rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/Belediye">
  <rdfs:label xml:lang="tr">Belediye</rdfs:label>
  <rdfs:subClassOf rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/Mahallildare"/>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/mahallelcerir"/>
      <owl:allValuesFrom rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/Mahalle"/>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim.owl"/>
</owl:Class>
```

<!-- http://araziyonetimi.org/ontoloji/idaribirim/BuyuksehirBelediyesi -->

```
<owl:Class rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/BuyuksehirBelediyesi">
  <rdfs:label xml:lang="tr">B y k ehir Belediyesi</rdfs:label>
  <rdfs:subClassOf rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/Belediye"/>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/ilcelcerir"/>
      <owl:allValuesFrom rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/BuyuksehirIlce"/>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/konumsaliliskiler/esittir"/>
      <owl:onClass rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/BuyuksehirIl"/>
      <owl:qualifiedCardinality rdf:datatype="xsd:nonNegativeInteger">1</owl:qualifiedCardinality>
    </owl:Restriction>
  </rdfs:subClassOf>
  <dc:description xml:lang="tr">B y k ehir olan illerdeki il belediyelerini ifade eder</dc:description>
  <rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim.owl"/>
</owl:Class>
```

<!-- http://araziyonetimi.org/ontoloji/idaribirim/BuyuksehirDisillce -->

```

<owl:Class rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/BuyuksehirDisillce">
  <rdfs:label xml:lang="tr">B y k ehir Dı ı İl e</rdfs:label>
  <rdfs:subClassOf rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/Ilce"/>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/illcindedir"/>
      <owl:onClass rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/BuyuksehirOlmayanIl"/>
      <owl:qualifiedCardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:qualifiedCardinality>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/koylcerir"/>
      <owl:allValuesFrom rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/Koy"/>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Restriction>
          <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/belediyelcerir"/>
          <owl:onClass rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/IlceBelediyesi"/>
          <owl:qualifiedCardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:qualifiedCardinality>
        </owl:Restriction>
        <owl:Restriction>
          <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/konumsaliliskiler/esittir"/>
          <owl:onClass rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/IlceBelediyesi"/>
          <owl:qualifiedCardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:qualifiedCardinality>
        </owl:Restriction>
      </owl:unionOf>
    </owl:Class>
  </rdfs:subClassOf>
  <owl:disjointWith rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/BuyuksehirIlce"/>
  <dc:description xml:lang="tr">B y k ehir olmayan illerdeki il eleri temsil eder.</dc:description>
  <rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim.owl"/>
</owl:Class>

```

```

<!-- http://araziyonetimi.org/ontoloji/idaribirim/BuyuksehirIl -->

```

```

<owl:Class rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/BuyuksehirIl">
  <rdfs:label xml:lang="tr">B y k ehir İl</rdfs:label>
  <rdfs:subClassOf rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/Il"/>

```

```

<rdfs:subClassOf>
  <owl:Restriction>
    <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/ilcelcerir"/>
    <owl:allValuesFrom rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/BuyuksehirIlce"/>
  </owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf>
  <owl:Restriction>
    <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/konumsaliliskiler/esittir"/>
    <owl:onClass rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/BuyuksehirBelediyesi"/>
    <owl:qualifiedCardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:qualifiedCardinality>
  </owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf>
  <owl:Restriction>
    <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/belediyeIcerir"/>
    <owl:someValuesFrom rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/BuyuksehirIlceBelediyesi"/>
  </owl:Restriction>
</rdfs:subClassOf>
<owl:disjointWith rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/BuyuksehirOlmayanIl"/>
<dc:description xml:lang="tr">B y k       olan illeri temsil eder.</dc:description>
<rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim.owl"/>
</owl:Class>

```

```

<!-- http://araziyonetimi.org/ontoloji/idaribirim/BuyuksehirIlce -->

```

```

<owl:Class rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/BuyuksehirIlce">
  <rdfs:label xml:lang="tr">B y k       İl e</rdfs:label>
  <rdfs:subClassOf rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/Ilce"/>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/belediyeIcerir"/>
      <owl:onClass rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/BuyuksehirIlceBelediyesi"/>
      <owl:qualifiedCardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:qualifiedCardinality>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/konumsaliliskiler/esittir"/>
      <owl:onClass rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/BuyuksehirIlceBelediyesi"/>
      <owl:qualifiedCardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:qualifiedCardinality>
    </owl:Restriction>
  </rdfs:subClassOf>

```

```

</rdfs:subClassOf>
<rdfs:subClassOf>
  <owl:Restriction>
    <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/illcindedir"/>
    <owl:onClass rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/BuyuksehirII"/>
    <owl:qualifiedCardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:qualifiedCardinality>
  </owl:Restriction>
</rdfs:subClassOf>
<dc:description xml:lang="tr">B y k ehir olan illerdeki il eleri temsil eder.</dc:description>
<rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim.owl"/>
</owl:Class>

<!-- http://araziyonetimi.org/ontoloji/idaribirim/BuyuksehirIlceBelediyesi -->

<owl:Class rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/BuyuksehirIlceBelediyesi">
  <rdfs:label xml:lang="tr">B y k ehir İl e Belediyesi</rdfs:label>
  <rdfs:subClassOf rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/Belediye"/>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/konumsaliliskiler/esittir"/>
      <owl:onClass rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/BuyuksehirIlce"/>
      <owl:qualifiedCardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:qualifiedCardinality>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/belediyeIcindedir"/>
      <owl:onClass rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/BuyuksehirBelediyesi"/>
      <owl:qualifiedCardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:qualifiedCardinality>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/illcindedir"/>
      <owl:onClass rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/BuyuksehirII"/>
      <owl:qualifiedCardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:qualifiedCardinality>
    </owl:Restriction>
  </rdfs:subClassOf>
  <dc:description xml:lang="tr">B y k ehir olan illerdeki il e belediyelerini ifade eder.</dc:description>
  <rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim.owl"/>
</owl:Class>

```

<!-- http://araziyonetimi.org/ontoloji/idaribirim/BuyuksehirOlmayanIl -->

```
<owl:Class rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/BuyuksehirOlmayanIl">
  <rdfs:label xml:lang="tr">B y k ehir Olmayan İl</rdfs:label>
  <rdfs:subClassOf rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/Il"/>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/belediyeIcerir"/>
      <owl:onClass rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/BeldeBelediyesi"/>
      <owl:minQualifiedCardinality rdf:datatype="&xsd;nonNegativeInteger">0</owl:minQualifiedCardinality>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/konumsaliliskiler/esittir"/>
      <owl:onClass rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/IlOzelIdaresi"/>
      <owl:qualifiedCardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:qualifiedCardinality>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/belediyeIcerir"/>
      <owl:onClass rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/IlBelediyesi"/>
      <owl:qualifiedCardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:qualifiedCardinality>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/koyIcerir"/>
      <owl:allValuesFrom rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/Koy"/>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/belediyeIcerir"/>
      <owl:someValuesFrom rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/IlceBelediyesi"/>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/ilcelcerir"/>
      <owl:allValuesFrom rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/BuyuksehirDisilce"/>
    </owl:Restriction>
  </rdfs:subClassOf>
```

```

    </owl:Restriction>
  </rdfs:subClassOf>
  <dc:description xml:lang="tr">Büyükşehir olmayan illeri temsil eder.</dc:description>
  <rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim.owl"/>
</owl:Class>

<!-- http://araziyonetimi.org/ontoloji/idaribirim/IdariBirim -->

<owl:Class rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/IdariBirim">
  <rdfs:label xml:lang="tr">İdari Birim</rdfs:label>
</owl:Class>

<!-- http://araziyonetimi.org/ontoloji/idaribirim/Il -->

<owl:Class rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/Il">
  <rdfs:label xml:lang="tr">İl</rdfs:label>
  <rdfs:subClassOf rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/MulkiIdare"/>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/mahallelcerir"/>
      <owl:allValuesFrom rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/Mahalle"/>
    </owl:Restriction>
  </rdfs:subClassOf>
  <owl:disjointWith rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/Ilce"/>
  <rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim.owl"/>
</owl:Class>

<!-- http://araziyonetimi.org/ontoloji/idaribirim/IlBelediyesi -->

<owl:Class rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/IlBelediyesi">
  <rdfs:label xml:lang="tr">İl Belediyesi</rdfs:label>
  <rdfs:subClassOf rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/Belediye"/>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/koylcerir"/>
      <owl:allValuesFrom rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/Koy"/>
    </owl:Restriction>
  </rdfs:subClassOf>

```

```

<rdfs:subClassOf>
  <owl:Restriction>
    <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/ilcelcerir"/>
    <owl:allValuesFrom rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/BuyuksehirDisillce"/>
  </owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf>
  <owl:Restriction>
    <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/illcindedir"/>
    <owl:onClass rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/BuyuksehirOlmayanIl"/>
    <owl:qualifiedCardinality rdf:datatype="xsd:nonNegativeInteger">1</owl:qualifiedCardinality>
  </owl:Restriction>
</rdfs:subClassOf>
<dc:description xml:lang="tr">Büyükşehir olmayan illerdeki il (merkez) belediyelerini ifade eder.</dc:description>
<rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim.owl"/>
</owl:Class>

```

```

<!-- http://araziyonetimi.org/ontoloji/idaribirim/IOzelIdaresi -->

```

```

<owl:Class rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/IOzelIdaresi">
  <rdfs:label xml:lang="tr">İl Özel İdaresi</rdfs:label>
  <rdfs:subClassOf rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/MahalliIdare"/>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/mahallelcerir"/>
      <owl:allValuesFrom rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/Mahalle"/>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/ilcelcerir"/>
      <owl:allValuesFrom rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/BuyuksehirDisillce"/>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/koylcerir"/>
      <owl:allValuesFrom rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/Koy"/>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
    <owl:Restriction>

```

```

        <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/konumsaliliskiler/esittir"/>
        <owl:onClass rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/BuyuksehirOlmayanIl"/>
        <owl:qualifiedCardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:qualifiedCardinality>
    </owl:Restriction>
</rdfs:subClassOf>
<rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim.owl"/>
</owl:Class>

```

```

<!-- http://araziyonetimi.org/ontoloji/idaribirim/Ilce -->

```

```

<owl:Class rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/Ilce">
    <rdfs:label xml:lang="tr">İlçe</rdfs:label>
    <rdfs:subClassOf rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/MulkiIdare"/>
    <rdfs:subClassOf>
        <owl:Restriction>
            <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/mahallelcerir"/>
            <owl:allValuesFrom rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/Mahalle"/>
        </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf>
        <owl:Restriction>
            <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/UAVTKodu"/>
            <owl:qualifiedCardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:qualifiedCardinality>
            <owl:onDataRange rdf:resource="&xsd;nonNegativeInteger"/>
        </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim.owl"/>
</owl:Class>

```

```

<!-- http://araziyonetimi.org/ontoloji/idaribirim/IlceBelediyesi -->

```

```

<owl:Class rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/IlceBelediyesi">
    <rdfs:label xml:lang="tr">İlçe Belediyesi</rdfs:label>
    <rdfs:subClassOf rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/Belediye"/>
    <rdfs:subClassOf>
        <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
                <owl:Restriction>
                    <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/ilcelcindedir"/>
                    <owl:onClass rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/BuyuksehirDisilce"/>

```

```

        <owl:qualifiedCardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:qualifiedCardinality>
    </owl:Restriction>
    <owl:Restriction>
        <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/konumsaliliskiler/esittir"/>
        <owl:onClass rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/BuyuksehirDisillce"/>
        <owl:qualifiedCardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:qualifiedCardinality>
    </owl:Restriction>
</owl:unionOf>
</owl:Class>
</rdfs:subClassOf>
<rdfs:subClassOf>
    <owl:Restriction>
        <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/illcindedir"/>
        <owl:onClass rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/BuyuksehirOlmayanIl"/>
        <owl:qualifiedCardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:qualifiedCardinality>
    </owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf>
    <owl:Restriction>
        <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/koylcerir"/>
        <owl:allValuesFrom rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/Koy"/>
    </owl:Restriction>
</rdfs:subClassOf>
<dc:description xml:lang="tr">Büyükşehir olmayan illerdeki ilçe belediyelerini ifade eder.</dc:description>
<rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim.owl"/>
</owl:Class>

<!-- http://araziyonetimi.org/ontoloji/idaribirim/Koy -->

<owl:Class rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/Koy">
    <rdfs:label xml:lang="tr">Köy</rdfs:label>
    <rdfs:subClassOf rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/MahalliIdare"/>
    <rdfs:subClassOf>
        <owl:Restriction>
            <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/mahallelcerir"/>
            <owl:onClass rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/Mahalle"/>
            <owl:minQualifiedCardinality rdf:datatype="&xsd;nonNegativeInteger">0</owl:minQualifiedCardinality>
        </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf>
        <owl:Restriction>
            <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/vikipedi"/>

```

```

        <owl:qualifiedCardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:qualifiedCardinality>
        <owl:onDataRange rdf:resource="&xsd:anyURI"/>
    </owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf>
    <owl:Restriction>
        <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/ilcelcindedir"/>
        <owl:onClass rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/BuyuksehirIlce"/>
        <owl:qualifiedCardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:qualifiedCardinality>
    </owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf>
    <owl:Restriction>
        <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/illcindedir"/>
        <owl:onClass rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/BuyuksehirOlmayanIl"/>
        <owl:qualifiedCardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:qualifiedCardinality>
    </owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf>
    <owl:Restriction>
        <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/alternatifAd"/>
        <owl:minQualifiedCardinality rdf:datatype="&xsd;nonNegativeInteger">0</owl:minQualifiedCardinality>
        <owl:onDataRange rdf:resource="&xsd:string"/>
    </owl:Restriction>
</rdfs:subClassOf>
<rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim.owl"/>
</owl:Class>

```

```

<!-- http://araziyonetimi.org/ontoloji/idaribirim/Mahalle -->

```

```

<owl:Class rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/Mahalle">
    <rdfs:label xml:lang="tr">Mahalle</rdfs:label>
    <rdfs:subClassOf rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/Mahallildare"/>
    <rdfs:subClassOf>
        <owl:Restriction>
            <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/sabitTanitimNumarasi"/>
            <owl:qualifiedCardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:qualifiedCardinality>
            <owl:onDataRange rdf:resource="&xsd;nonNegativeInteger"/>
        </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf>
        <owl:Restriction>

```

```

    <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/illcindedir"/>
    <owl:onClass rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/Il"/>
    <owl:qualifiedCardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:qualifiedCardinality>
  </owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf>
  <owl:Restriction>
    <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/belediyeIcindedir"/>
    <owl:someValuesFrom rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/Belediye"/>
  </owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf>
  <owl:Restriction>
    <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/wikipedi"/>
    <owl:qualifiedCardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:qualifiedCardinality>
    <owl:onDataRange rdf:resource="&xsd:anyURI"/>
  </owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf>
  <owl:Restriction>
    <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/ilOzelIdareIcindedir"/>
    <owl:onClass rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/IlOzelIdaresi"/>
    <owl:minQualifiedCardinality rdf:datatype="&xsd;nonNegativeInteger">0</owl:minQualifiedCardinality>
  </owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf>
  <owl:Restriction>
    <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/alternatifAd"/>
    <owl:minQualifiedCardinality rdf:datatype="&xsd;nonNegativeInteger">0</owl:minQualifiedCardinality>
    <owl:onDataRange rdf:resource="&xsd:string"/>
  </owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf>
  <owl:Class>
    <owl:unionOf rdf:parseType="Collection">
      <owl:Restriction>
        <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/koyIcindedir"/>
        <owl:onClass rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/Koy"/>
        <owl:qualifiedCardinality rdf:datatype="&xsd;nonNegativeInteger">0</owl:qualifiedCardinality>
      </owl:Restriction>
      <owl:Restriction>
        <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/koyIcindedir"/>
        <owl:onClass rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/Koy"/>
        <owl:qualifiedCardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:qualifiedCardinality>
      </owl:Restriction>
    </owl:unionOf>
  </owl:Class>

```

```

        </owl:Restriction>
    </owl:unionOf>
</owl:Class>
</rdfs:subClassOf>
<rdfs:subClassOf>
    <owl:Restriction>
        <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/ilcelcindedir"/>
        <owl:onClass rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/ilce"/>
        <owl:qualifiedCardinality rdf:datatype="xsd:nonNegativeInteger">1</owl:qualifiedCardinality>
    </owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf>
    <owl:Restriction>
        <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/postaKodu"/>
        <owl:qualifiedCardinality rdf:datatype="xsd:nonNegativeInteger">1</owl:qualifiedCardinality>
        <owl:onDataRange rdf:resource="xsd:nonNegativeInteger"/>
    </owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf>
    <owl:Restriction>
        <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/UAVTKodu"/>
        <owl:qualifiedCardinality rdf:datatype="xsd:nonNegativeInteger">1</owl:qualifiedCardinality>
        <owl:onDataRange rdf:resource="xsd:nonNegativeInteger"/>
    </owl:Restriction>
</rdfs:subClassOf>
<rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim.owl"/>
</owl:Class>

```

```

<!-- http://araziyonetimi.org/ontoloji/idaribirim/MahalliIdare -->

```

```

<owl:Class rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/MahalliIdare">
    <rdfs:label xml:lang="tr">Mahalli idare</rdfs:label>
    <rdfs:subClassOf rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/IdariBirim"/>
    <rdfs:subClassOf>
        <owl:Restriction>
            <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/harita"/>
            <owl:qualifiedCardinality rdf:datatype="xsd:nonNegativeInteger">1</owl:qualifiedCardinality>
            <owl:onDataRange rdf:resource="xsd:anyURI"/>
        </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf>
        <owl:Restriction>

```

```

    <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/yuzolcum"/>
    <owl:qualifiedCardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:qualifiedCardinality>
    <owl:onDataRange rdf:resource="&xsd;decimal"/>
  </owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf>
  <owl:Restriction>
    <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/kurulusTarihi"/>
    <owl:qualifiedCardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:qualifiedCardinality>
    <owl:onDataRange rdf:resource="&xsd;date"/>
  </owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf>
  <owl:Restriction>
    <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/geometri/koordinat"/>
    <owl:someValuesFrom rdf:resource="http://araziyonetimi.org/ontoloji/geometri/Geometri"/>
  </owl:Restriction>
</rdfs:subClassOf>
<owl:disjointWith rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/Mulkildare"/>
<rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim.owl"/>
</owl:Class>

<!-- http://araziyonetimi.org/ontoloji/idaribirim/Mulkildare -->

<owl:Class rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/Mulkildare">
  <rdfs:label xml:lang="tr">Mülki İdare</rdfs:label>
  <rdfs:subClassOf rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/IdariBirim"/>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/alternatifAd"/>
      <owl:minQualifiedCardinality rdf:datatype="&xsd;nonNegativeInteger">0</owl:minQualifiedCardinality>
      <owl:onDataRange rdf:resource="&xsd;string"/>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/kurulusTarihi"/>
      <owl:qualifiedCardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:qualifiedCardinality>
      <owl:onDataRange rdf:resource="&xsd;date"/>
    </owl:Restriction>
  </rdfs:subClassOf>
</rdfs:subClassOf>

```

```

<owl:Restriction>
  <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/geometri/koordinat"/>
  <owl:someValuesFrom rdf:resource="http://araziyonetimi.org/ontoloji/geometri/Geometri"/>
</owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf>
  <owl:Restriction>
    <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/yuzolcum"/>
    <owl:qualifiedCardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:qualifiedCardinality>
    <owl:onDataRange rdf:resource="&xsd;decimal"/>
  </owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf>
  <owl:Restriction>
    <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/nufus"/>
    <owl:qualifiedCardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:qualifiedCardinality>
    <owl:onDataRange rdf:resource="&xsd;nonNegativeInteger"/>
  </owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf>
  <owl:Restriction>
    <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/wikipedi"/>
    <owl:qualifiedCardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:qualifiedCardinality>
    <owl:onDataRange rdf:resource="&xsd;anyURI"/>
  </owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf>
  <owl:Restriction>
    <owl:onProperty rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/harita"/>
    <owl:qualifiedCardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:qualifiedCardinality>
    <owl:onDataRange rdf:resource="&xsd;anyURI"/>
  </owl:Restriction>
</rdfs:subClassOf>
<rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim.owl"/>
</owl:Class>

<!-- http://www.w3.org/2002/07/owl#Thing -->

<owl:Class rdf:about="&owl;Thing"/>

```

```

<!--
////////////////////////////////////
//
// Individuals
//
////////////////////////////////////
-->

```

```

<!-- http://araziyonetimi.org/ontoloji/idaribirim/ -->

```

```

<owl:Thing rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/">
  <rdf:type rdf:resource="&owl;NamedIndividual"/>
</owl:Thing>

```

```

<!--
////////////////////////////////////
//
// General axioms
//
////////////////////////////////////
-->

```

```

<rdf:Description>
  <rdf:type rdf:resource="&owl;AllDisjointClasses"/>
  <owl:members rdf:parseType="Collection">
    <rdf:Description rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/BeldeBelediyesi"/>
    <rdf:Description rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/BuyuksehirBelediyesi"/>
    <rdf:Description rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/BuyuksehirIlceBelediyesi"/>
    <rdf:Description rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/IlBelediyesi"/>
    <rdf:Description rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/IlceBelediyesi"/>
  </owl:members>
</rdf:Description>
<rdf:Description>
  <rdf:type rdf:resource="&owl;AllDisjointClasses"/>
  <owl:members rdf:parseType="Collection">
    <rdf:Description rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/Belediye"/>
    <rdf:Description rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/IIOzelIdaresi"/>
    <rdf:Description rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/Koy"/>
    <rdf:Description rdf:about="http://araziyonetimi.org/ontoloji/idaribirim/Mahalle"/>
  </owl:members>
</rdf:Description>

```

```
</owl:members>
</rdf:Description>
</rdf:RDF>
```

```
<!-- Generated by the OWL API (version 3.4.2) http://owlapi.sourceforge.net -->
```

İDARİ BİRİM RDF ÖRNEKLERİ

• Abbasağa RDF/ XML Örneği

```

<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF
  xmlns:cc="http://creativecommons.org/ns#"
  xmlns:dcterms="http://purl.org/dc/terms/"
  xmlns:foaf="http://xmlns.com/foaf/0.1/"
  xmlns:idaribirim="http://araziyonetimi.org/ontoloji/idaribirim/"
  xmlns:skos="http://www.w3.org/2004/02/skos/core#"
  xmlns:konumsaliliskiler="http://araziyonetimi.org/ontoloji/konumsaliliskiler/"
  xmlns:georss="http://www.georss.org/georss/"
  xmlns:geometri="http://araziyonetimi.org/ontoloji/geometri/"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:wgs84_pos="http://www.w3.org/2003/01/geo/wgs84_pos#"
>
<rdf:Description rdf:about="http://araziyonetimi.org/id/idaribirim/istanbul-besiktas-abbasaga">
  <rdf:type rdf:resource="http://araziyonetimi.org/ontoloji/idaribirim/Mahalle"/>
  <idaribirim:postaKodu>34353</idaribirim:postaKodu>
  <idaribirim:belediyeIcindedir rdf:resource="http://araziyonetimi.org/id/idaribirim/istanbulbb-besiktasb"/>
  <idaribirim:vikipedi rdf:resource="http://tr.wikipedia.org/wiki/Abbasa%C4%9Fa_Be%C5%9Fikta%C5%9F"/>
  <idaribirim:belediyeIcindedir rdf:resource="http://araziyonetimi.org/id/idaribirim/istanbulbb"/>
  <idaribirim:kurulusTarihi rdf:datatype="http://www.w3.org/2001/XMLSchema#date">1970-01-01</idaribirim:kurulusTarihi>
  <rdfs:isDefinedBy rdf:resource="http://araziyonetimi.org/id/idaribirim/istanbul-besiktas-abbasaga.rdf"/>
  <idaribirim:UAVTKodu>40219</idaribirim:UAVTKodu>
  <idaribirim:alternatifAd>Abbasağa</idaribirim:alternatifAd>
  <wgs84_pos:lat>41.05125</wgs84_pos:lat>
  <idaribirim:ilcelcindedir rdf:resource="http://araziyonetimi.org/id/idaribirim/istanbul-besiktas"/>
  <idaribirim:illcindedir rdf:resource="http://araziyonetimi.org/id/idaribirim/istanbul"/>
  <konumsaliliskiler:icindedir rdf:resource="http://araziyonetimi.org/id/idaribirim/istanbul-besiktas"/>
  <wgs84_pos:long>29.00555</wgs84_pos:long>
  <konumsaliliskiler:icindedir rdf:resource="http://araziyonetimi.org/id/idaribirim/istanbul"/>
  <idaribirim:sabitTanitimNumarasi>1</idaribirim:sabitTanitimNumarasi>
  <georss:point>41.05125 29.00555</georss:point>
  <idaribirim:harita>1</idaribirim:harita>
  <idaribirim:yuzolcum>143932.141</idaribirim:yuzolcum>
  <konumsaliliskiler:icindedir rdf:resource="http://araziyonetimi.org/id/idaribirim/istanbulbb-besiktasb"/>
  <konumsaliliskiler:dokunur rdf:resource="http://araziyonetimi.org/id/idaribirim/istanbul-besiktas-ciannuma"/>
  <konumsaliliskiler:dokunur rdf:resource="http://araziyonetimi.org/id/idaribirim/istanbul-besiktas-turkali"/>
  <idaribirim:nufus>5465</idaribirim:nufus>
  <skos:prefLabel>Abbasağa</skos:prefLabel>
  <rdfs:label>Abbasağa</rdfs:label>
  <konumsaliliskiler:icindedir rdf:resource="http://araziyonetimi.org/id/idaribirim/istanbulbb"/>
  <geometri:koordinat rdf:resource="http://araziyonetimi.org/id/idaribirim/geometri/istanbul-besiktas-abbasaga"/>
  <konumsaliliskiler:dokunur rdf:resource="http://araziyonetimi.org/id/idaribirim/istanbul-besiktas-dikikitas"/>
</rdf:Description>
<rdf:Description rdf:about="http://araziyonetimi.org/doc/idaribirim/istanbul-besiktas-abbasaga.rdf">
  <cc:attributionName rdf:datatype="http://www.w3.org/2001/XMLSchema#string">YTU Harita Kamu</cc:attributionName>
  <dcterms:description rdf:datatype="http://www.w3.org/2001/XMLSchema#string">İdari Birim</dcterms:description>
  <cc:attributionURL rdf:resource="http://araziyonetimi.org/id/idaribirim/istanbul-besiktas-abbasaga"/>

```

```

<dcterms:modified rdf:datatype="http://www.w3.org/2001/XMLSchema#date">2014-04-30</dcterms:modified>
<foaf:primaryTopic rdf:resource="http://araziyonetimi.org/id/idaribirim/istanbul-besiktas-abbasaga"/>
<dcterms:contributor>Abdullah Kara</dcterms:contributor>
<rdf:type rdf:resource="http://xmlns.com/foaf/0.1/Document"/>
<cc:license rdf:resource="http://creativecommons.org/licenses/by/4.0/" />
<dcterms:created rdf:datatype="http://www.w3.org/2001/XMLSchema#date">2014-04-30</dcterms:created>
<dcterms:title rdf:resource="http://araziyonetimi.org/id/idaribirim/istanbul-besiktas-abbasaga"/>
</rdf:Description>
</rdf:RDF>

```

• Turtle Abbasğa Örneđi

```

@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix wgs84_pos: <http://www.w3.org/2003/01/geo/wgs84_pos#> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix dcterms: <http://purl.org/dc/terms/> .
@prefix owl: <http://www.w3.org/2002/07/owl#> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix cc: <http://creativecommons.org/ns#> .

<http://araziyonetimi.org/id/idaribirim/istanbul-besiktas-abbasaga> a <http://araziyonetimi.org/ontoloji/idaribirim/Mahalle> ;
rdfs:label "Abbasaga" ;
rdfs:isDefinedBy <http://araziyonetimi.org/id/idaribirim/istanbul-besiktas-abbasaga.ttl> ;
<http://araziyonetimi.org/ontoloji/idaribirim/alternatifAd> "Abbasaga" ;
<http://araziyonetimi.org/ontoloji/idaribirim/nufus> "5465" ;
<http://araziyonetimi.org/ontoloji/idaribirim/yuzolcum> "143932.141" ;
<http://araziyonetimi.org/ontoloji/idaribirim/UAVTKodu> "40219" ;
<http://araziyonetimi.org/ontoloji/idaribirim/postaKodu> "34353" ;
<http://araziyonetimi.org/ontoloji/idaribirim/sabitTanitimNumarasi> "1" ;
<http://araziyonetimi.org/ontoloji/idaribirim/illcindedir>
<http://araziyonetimi.org/id/idaribirim/istanbul> ;
<http://araziyonetimi.org/ontoloji/idaribirim/ilcelcindedir>
<http://araziyonetimi.org/id/idaribirim/istanbul-besiktas> ;
<http://araziyonetimi.org/ontoloji/idaribirim/belediyelcindedir>
<http://araziyonetimi.org/id/idaribirim/istanbulbb> , <http://araziyonetimi.org/id/idaribirim/istanbulbb-besiktasb> ;
<http://araziyonetimi.org/ontoloji/geometri/koordinat>
<http://araziyonetimi.org/id/idaribirim/geometri/istanbul-besiktas-abbasaga> ;
<http://araziyonetimi.org/ontoloji/konumsaliliskiler/icindedir>
<http://araziyonetimi.org/id/idaribirim/istanbul> , <http://araziyonetimi.org/id/idaribirim/istanbul-besiktas> ,
<http://araziyonetimi.org/id/idaribirim/istanbulbb> ,
<http://araziyonetimi.org/id/idaribirim/istanbulbb-besiktasb> ;
<http://araziyonetimi.org/ontoloji/konumsaliliskiler/dokunur>
<http://araziyonetimi.org/id/idaribirim/istanbul-besiktas-kihannuma> , <http://araziyonetimi.org/id/idaribirim/istanbul-besiktas-
dikikitas> ,
<http://araziyonetimi.org/id/idaribirim/istanbul-besiktas-turkali> ;
wgs84_pos:lat "41.05125" ;
wgs84_pos:long "29.00555" ;
<http://www.georss.org/georss/point> "41.05125 29.00555" ;
<http://www.w3.org/2004/02/skos/core#prefLabel> "Abbasaga" ;
<http://araziyonetimi.org/ontoloji/idaribirim/kurulusTarihi> "1970-01-01"^^xsd:date ;
<http://araziyonetimi.org/ontoloji/idaribirim/vikipedi> <http://tr.wikipedia.org/wiki/Abbasaga%20C4%9Fa,_Be%C5%9Fika%C5%9F> ;
<http://araziyonetimi.org/ontoloji/idaribirim/harita> "1" .
<http://araziyonetimi.org/doc/idaribirim/istanbul-besiktas-abbasaga.ttl> a foaf:Document ;
dcterms:title <http://araziyonetimi.org/id/idaribirim/istanbul-besiktas-abbasaga> ;
dcterms:created "2014-04-30"^^xsd:date ;
dcterms:modified "2014-04-30"^^xsd:date ;
foaf:primaryTopic <http://araziyonetimi.org/id/idaribirim/istanbul-besiktas-abbasaga> ;
dcterms:contributor "Abdullah Kara" ;
dcterms:description "Idari Birim"^^xsd:string ;
cc:attributionName "YTU Harita Kamu"^^xsd:string ;
cc:attributionURL <http://araziyonetimi.org/id/idaribirim/istanbul-besiktas-abbasaga> ;
cc:license <http://creativecommons.org/licenses/by/4.0/> .

```

VOID SÖZLÜĞÜ

@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix dcterms: <http://purl.org/dc/terms/> .
@prefix void: <http://rdfs.org/ns/void#> .

<http://araziyonetimi.org/doc/> rdf:type void:Dataset ;
foaf:homepage <http://araziyonetimi.org/> ;
dcterms:title "Idari Birim" ;
dcterms:description "Idari birimler yer adlari dizini" ;
dcterms:publisher <http://www.yarbis.yildiz.edu.tr/abkara> ;
dcterms:license <http://creativecommons.org/licenses/by-sa/3.0/> ;
void:sparqlEndpoint <http://araziyonetimi.org/sparql> ;
void:uriLookupEndpoint <http://araziyonetimi.org/sparql> ;
void:vocabulary <http://purl.org/dc/terms/> ;
void:vocabulary <http://www.w3.org/2000/01/rdf-schema#> ;
void:vocabulary <http://www.w3.org/2003/01/geo/wgs84_pos#> ;
void:vocabulary <http://xmlns.com/foaf/0.1/> ;
void:vocabulary <http://www.w3.org/2002/07/owl#> ;
void:vocabulary <http://www.w3.org/2001/XMLSchema#> ;
void:vocabulary <http://www.w3.org/1999/02/22-rdf-syntax-ns#> ;
void:vocabulary <http://creativecommons.org/ns#> ;
void:vocabulary <http://araziyonetimi.org/ontoloji/idaribirim.owl> ;
void:vocabulary <http://araziyonetimi.org/ontoloji/konumsaliliskiler.owl> ;
void:vocabulary <http://araziyonetimi.org/ontoloji/geometri.owl> ;
void:exampleResource <http://araziyonetimi.org/doc/idaribirim/istanbul-besiktas-abbasaga> .

SPARQL SORGU ÖRNEKLERİ

1)

```
SELECT * WHERE { ?s ?p ?o }
```

sorgusu tüm RDF ifadelerine erişmek için kullanılır.

2)

```
PREFIX konumsaliliskiler: <http://www.araziyonetimi.org/ontoloji/konumsaliliskiler/>
```

```
SELECT *
```

```
WHERE
```

```
{ ?s konumsaliliskiler:icerir ?o }
```

sorgusu, idari birimler veri setinde içerir konumsal ilişkisiyle birbiriyle ilişkili olan idari birim detaylarına erişim sağlar.

3)

```
PREFIX dc: <http://purl.org/dc/elements/1.1/>
```

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
```

```
PREFIX apf: <http://jena.hpl.hp.com/ARQ/property#>
```

```
PREFIX idaribirim: <http://araziyonetimi.org/ontoloji/idaribirim/>
```

```
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
```

```
PREFIX owl: <http://www.w3.org/2002/07/owl#>
```

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
```

```
PREFIX fn: <http://www.w3.org/2005/xpath-functions#>
```

```
PREFIX konumsaliliskiler: <http://araziyonetimi.org/ontoloji/konumsaliliskiler/>
```

```
SELECT ?s ?o
```

```
WHERE
```

```
{ ?s konumsaliliskiler:dokunur ?o }
```

```
ORDER BY ?s
```

sorgusu, birbirine dokunan idari birimleri (özne ve nesne olarak) seçilip, özneye göre sıralanmasını sağlar.

ÖZGEÇMİŞ

KİŞİSEL BİLGİLER

Adı Soyadı : Abdullah KARA
Doğum Tarihi ve Yeri : 20.03.1987, Altındağ
Yabancı Dili : İngilizce
E-posta : abkara@yildiz.edu.tr

ÖĞRENİM DURUMU

Derece	Alan	Okul/Üniversite	Mezuniyet Yılı
Lisans	Geomatik Mühendisliği	İstanbul Teknik Üniversitesi	2012
Lise	Fen	Ankara Çankaya Atatürk Anadolu Lisesi	2005

İŞ TECRÜBESİ

Yıl	Firma/Kurum	Görevi
2011 -2013	İTU Teknokent	Mühendis

YAYINLARI

Makale

- 1.Kara, A. ve Çağdaş, V. (2014). “Türkiye İdari Birimler Dizini İçin Bir Ontoloji Modeli Önerisi”, Sigma Mühendislik ve Fen Bilimleri Dergisi, 32(3), 349-363.