

# ONTOLOJİ TABANLI BİR KİTAP SORGULAMA SİSTEMİ GERÇEKLEŞTİRİMİ

Ayşegül ALAYBEYOĞLU, Aybars UĞUR

Ege Üniversitesi, Bilgisayar Mühendisliği Bölümü, 35100, Bornova, İZMİR  
{[aysegul.alaybeyoglu](mailto:aysegul.alaybeyoglu@ege.edu.tr), [aybars.ugur](mailto:aybars.ugur@ege.edu.tr)}@ege.edu.tr

## ÖZET

Bu bildiride, RDF tabanlı bir kitap ontolojisinin hazırlanması ve ontolojinin anlamsal web ortamında kullanılmasına yönelik gerçekleştirilen bir çalışma anlatılmıştır. RDF tabanlı OWL dili ile ontoloji geliştirilmiştir. Bu ontolojide kitaplar birer RDF kaynağı olacak şekilde tasarlanmışlardır. Bu kaynaklara ait özelliklerde yine ilgili OWL yapıları kullanılarak ontoloji bünyesinde yer almıştır. İlgili ontoloji üzerinde RDQL kullanılarak sorgu işleten ve JENA API'yi kullanan bir sistem yazılımı da tasarlanıp hayata geçirilmiştir. Böylelikle ontoloji bir RDF kaynak modeli olarak yazılım bileşenleri tarafından işlenebilmekte ve onun üzerinde tıpkı veritabanı sistemlerindeki sorgulama yaklaşımına benzer bir biçimde sorgular işletilerek ilgili RDF kaynaklarına erişim sağlanabilmektedir. Elde edilen bu kaynaklar daha sonra anlamsal web ortamında ilgili yapılar tarafından kullanılabilirler.

## ABSTRACT

In this paper, development and use of an RDF based book ontology within the Semantic Web environment are discussed. In the developed ontology each book item has been designed as an RDF Resource. Likewise, attributes of each resource have been included within the ontology by using proper OWL constructs. A software has also been designed and implemented in order to process and query on the related ontology by using JENA API and RDQL. Hence ontology can be processed by software components as an RDF resource model and can be queried as if it is an ordinary database. Ontology resources that are obtained by execution of appropriate queries, can be used by related components on Semantic Web environment.

**Anahtar Kelimeler:** Anlamsal Web, Ontoloji, RDF.

## 1. GİRİŞ

Günümüzdeki web içeriğine temel oluşturan bir çok yapıya bakıldığında bu yapıların insanların okuması, anlaması ve kullanımı için tasarlanmış ve geliştirilmiş olduğu görülür. Mevcut web alt yapısında sunulan içeriğin bilgisayarca okunur ve bilgisayarca anlaşılabilir olması çok zordur fakat bunun sağlanabilmesi için de yeni bir modele gereksinim duyulmuştur. Bu yeni web modeli de "Anlamsal Web" 'tir [1].

Anlamsal Web kavramı ilk olarak Tim Berners Lee tarafından öne sürülmüştür. Anlamsal Web yeni ve ayrı bir web olmayıp,

bilgilere iyi tanımlanmış anlamların verildiği, bilgisayarların ve insanların birlikte çalışmalarına imkan veren bugünkü web'in bir uzantısıdır[2]. Buradaki temel amaç iyi tanımlanmış ve ilişkilendirilmiş olan bilgilerin ve servislerin web ortamında kolay bir şekilde bilgisayarca-okunabilir ve bilgisayarca-anlaşılabilir olmasını sağlayacak standartların ve teknolojilerin geliştirilmesidir[3].

Bu çalışmada da Anlamsal Web'in temel bileşeni olan ontoloji'den RDF tabanlı bir sorgulama dili olan RDQL ile sorgulama gerçekleştirimi çalışması anlatılmıştır. Bildirinin girişinde anlamsal web ile ilgili bilgiler verildikten sonra devamında

uygulamamızın temelini oluşturan ontoloji, ontoloji dilleri, JENA ve RDQL teknolojilerinden bahsedilmiştir. Bir sonraki bölümde de uygulamamızın gerçekleştirimi hakkında bilgiler verilmiştir. Uygulamamızda kitap domaini için bir ontoloji hazırlanmıştır. Daha sonrada ontolojimizde anlamsal olarak tuttuğumuz bilgileri RDQL ile sorgulamış ve sorgumuza karşılık gelen cevabı elde etmiş bulunmaktayız. Sonuç bölümünde ise gerçekleştirdiğimiz uygulamanın sonuçları değerlendirilmiş ve gelecekte hedeflenen çalışmalardan bahsedilmiştir.

## 2. SİSTEMİN ALT YAPISINI OLUŞTURAN TEKNOLOJİLER

### *Ontoloji*

Anlamsal Web'in en temel bileşeni olan ontoloji, kavramlar arasındaki ilişkileri formal olarak içeren bir dokümandır. Sözlük anlamı "varlık bilimi" olarak tanımlanan ontolojilerin bilgisayar biliminde en çok kabul gören tanımı ise "kavramsallaştırmanın biçimsel ve açıkça belirtilmesi"dir[1]. Burada kavramsallaştırma ile kast edilen, belirli bir ön alanda, bu ön alana ait soyut model oluşturma anlamına gelmektedir. Ontolojiler herhangi bir alanda standart olarak kullanılacak ortak ve paylaşılan sözcük kümelerini (vocabulary) veya terminolojiyi belirler.

### *Ontoloji Dilleri (OWL ve RDF)*

Ontolojiler, ontoloji dilleri ile tanımlanmaktadır. Bir çok alan için değişik ontoloji dillerinde ontolojiler geliştirilmektedir. Bu dillerden en çok kullanılanları RDF (Resource Description Framework) ve OWL (Web Ontology Language)'dur. RDF'de her kaynağın bir URI'ye sahip olması ile ve nesne, fiil, özne üçlüsünün yer alması, anlamın ifade edilmesini sağlamaktadır. İlgili üçlüler de XML tag'ları ile ifade edilmektedirler[2]. RDFS (RDF Schema) gösterimi, RDF veri modelini genişleten bir tip sistemidir. Bu tip sistemi, bir alanda kullanılacak olan sözcük kümesini tanımlar. Bu sözcük kümesi de bir alanda kullanılacak olan nesnelere, nesnelere

arasındaki alt/üst küme ilişkileri, özellikler, özelliklerin alabileceği değerleri tanımlar.[4]. Bir başka çok sık kullanılan ontoloji dili ise OWL (Web Ontology Language)'dur. OWL dili RDF diline bir eklentidir ve OWL Full, OWL DL, OWL Lite olmak üzere üçe ayrılmıştır. OWL dili ile özellikler, nesne ve veri tipi olmak üzere ikiye ayrılmış, küme işlemleri getirilmiş, özelliklerin alabileceği değerler üzerinde kısıtlamalar koyabilme ve eşleme ilişkileri tanımlayabilme sağlanmıştır [6].

### *Sorgulama*

Ontoloji oluşturulduktan sonra, yapılabilecek işlemlerden birisi de ontolojide tutulan bilgilerin bir ontoloji sorgulama dili ile sorgulanmasıdır. RDF tabanlı herhangi bir ontolojinin Java kullanılarak modelinin oluşturulup üzerinde sorgulamanın ve çıkarsamanın yapılabilmesi için JENA çerçevesi kullanılabilir. Bu çalışmada da ontolojimiz RDF tabanlı olduğu için RDF tabanlı bir sorgulama diline ihtiyaç vardır. Bunun için de JENA yazılım çerçevesi içerisinde yer alan RDQL sorgulama dili [10] kullanılmıştır.

## 3. SİSTEM

### *Ontolojinin Oluşturulması*

Ontoloji tabanlı bir sistem hazırlandığı için kaynakları tanımlayan bir ontolojinin olması gerekmektedir. Bu nedenle bu çalışma kapsamında bir ontoloji geliştirilmiştir. Sistem, kitap sorgulama ile ilgili olduğu için ontoloji Kitap kaynaklarını ve onların sahip olduğu özellikleri (adı, yazarları, basım yılı, vb) içermektedir. Ontolojiyi tasarlama ve hayata geçirmede kullanılan ontoloji dili Ontoloji Web Language (OWL)'dir. Haliyle yukarıda bahsedilen kaynaklar ontolojide birer Owl Individual'ı olarak yer almaktadır. Ontolojiyi geliştirmek için OWL'ın kullanılmasının nedeni bu dilin W3C (World Wide Web Consortium) tarafından standart kabul edilmesidir.

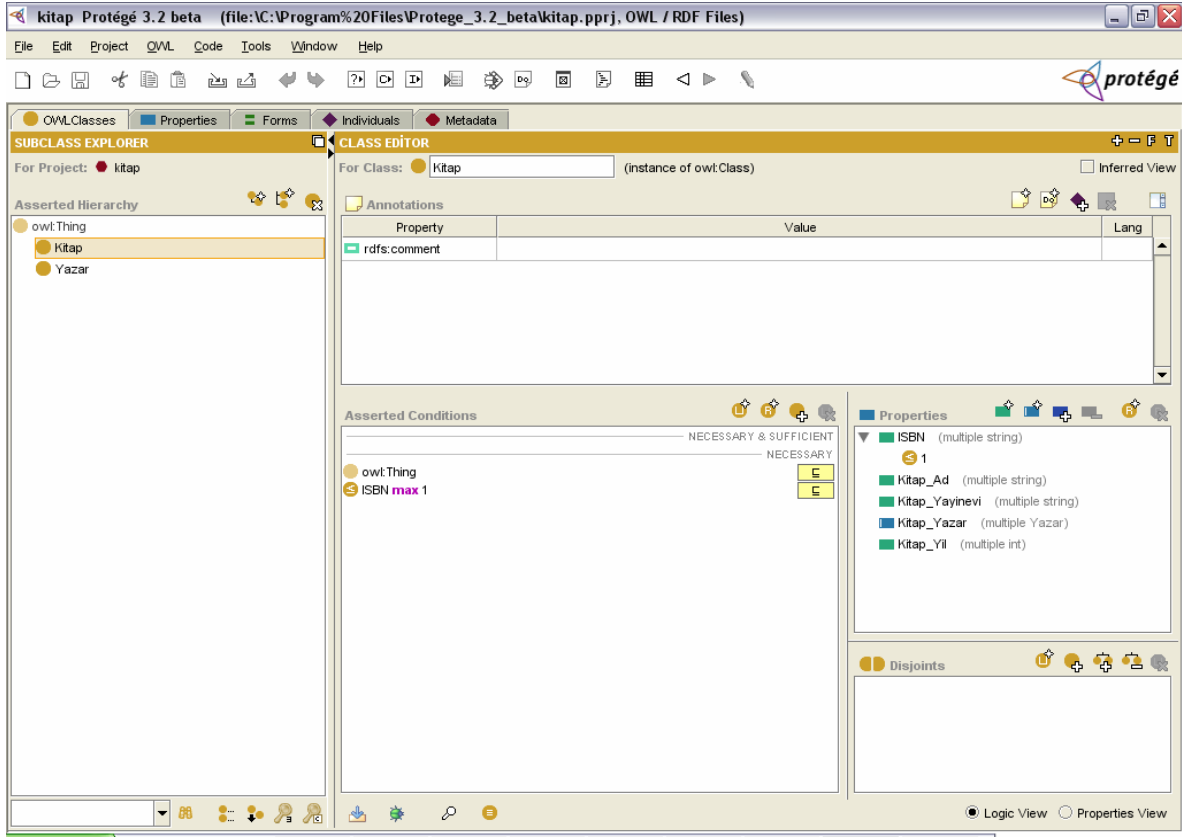
Ontolojiyi oluşturmak için Protege [11] ontoloji geliştirme editörü kullanılmıştır. Protege, sistem geliştiricileri ve uzmanları

tarafından bilgi tabanlı sistemlerin geliştirilmesi amacıyla kullanılan entegre bir yazılım aracıdır.

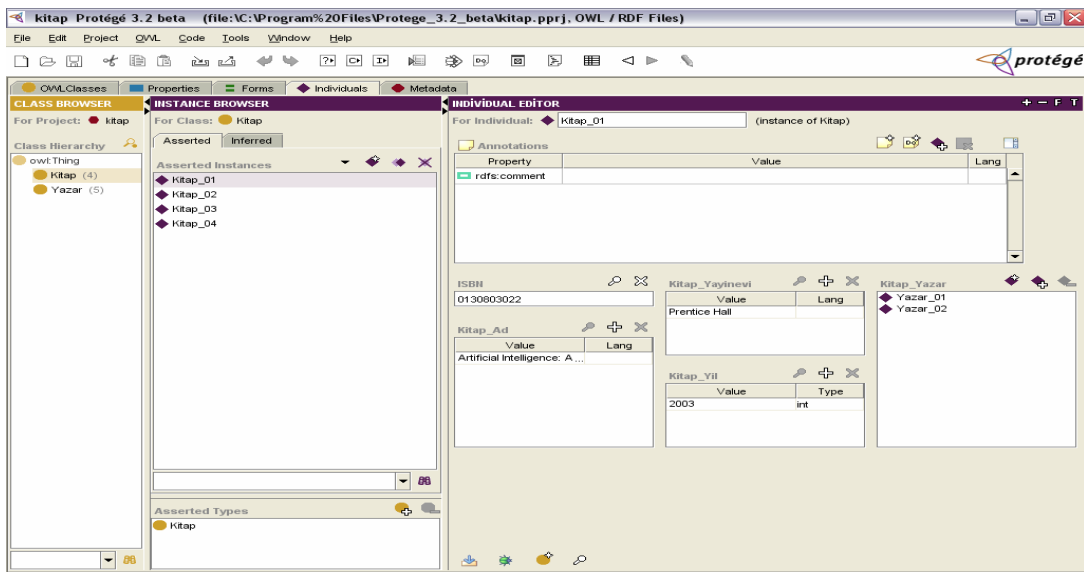
Şekil 1’de kitap ontolojisinin geliştirilmesi sırasında kullanılan Protege 3.2’nin grafiksel kullanıcı ara yüzü verilmiştir.

### Ontolojide Individualların Oluşturulması

Bu ontolojiye bağlı olarak hazırlanan kitap kaynakları (individualları) da yine editör aracılığıyla Şekil 2 ‘de de gösterildiği gibi ontolojiye dahil edilmiştir .



Şekil 1: Protege 3.2 editöründe ontolojinin geliştirilmesi sırasında elde edilen ekran görüntüsü



Şekil 2: Ontolojideki individualların (kitap kaynaklarının) Protege individual editörü kullanılarak hazırlanması

## Ontoloji OWL kodu

Hazırlanan ontolojide kitap kaynakları ve bu kaynaklara ait özellikler (ISBN, Ad, Yayınevi, Basım yılı ve yazarları) yer almıştır. Kitap kaynakları haricinde

yazarlar da ontoloji de birer OWL Individual'ı olarak bulunmuşlardır. Aşağıda sistemin ontolojisinden bir kesit verilmiştir:

```
<?xml version="1.0"?>
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:owl="http://www.w3.org/2002/07/owl#"
  xmlns="http://alaybeyoglu.googlepages.com/kitaplar.owl#"
  xml:base="http://alaybeyoglu.googlepages.com/kitaplar.owl">
  <owl:Ontology rdf:about=""/>
  <owl:Class rdf:ID="Yazar"/>
  <owl:Class rdf:ID="Kitap">
    <rdfs:subClassOf rdf:resource="http://www.w3.org/2002/07/owl#Thing"/>
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:maxCardinality
rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
          >1</owl:maxCardinality>
        <owl:onProperty>
          <owl:DatatypeProperty rdf:ID="ISBN"/>
        </owl:onProperty>
      </owl:Restriction>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:ObjectProperty rdf:ID="Kitap_Yazar">
    <rdfs:domain rdf:resource="#Kitap"/>
    <rdfs:range rdf:resource="#Yazar"/>
  </owl:ObjectProperty>
  <owl:DatatypeProperty rdf:about="#ISBN">
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    <rdfs:domain rdf:resource="#Kitap"/>
  </owl:DatatypeProperty>
  <owl:DatatypeProperty rdf:ID="Kitap_Ad">
    <rdfs:domain rdf:resource="#Kitap"/>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
  </owl:DatatypeProperty>
  <owl:DatatypeProperty rdf:ID="Isim">
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    <rdfs:domain rdf:resource="#Yazar"/>
  </owl:DatatypeProperty>
  <owl:DatatypeProperty rdf:ID="Kitap_Yil">
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#int"/>
    <rdfs:domain rdf:resource="#Kitap"/>
  </owl:DatatypeProperty>
  <owl:DatatypeProperty rdf:ID="Soyisim">
    <rdfs:domain rdf:resource="#Yazar"/>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
  </owl:DatatypeProperty>
  <owl:DatatypeProperty rdf:ID="Email">
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    <rdfs:domain rdf:resource="#Yazar"/>
  </owl:DatatypeProperty>
  <owl:DatatypeProperty rdf:ID="Kitap_Yayınevi">
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    <rdfs:domain rdf:resource="#Kitap"/>
  </owl:DatatypeProperty>
</rdf:RDF>
```

```

</owl:DatatypeProperty>

<Yazar rdf:ID="Yazar_01">
  <Email rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  ></Email>
  <Soyisim rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >Russell</Soyisim>
  <Isim rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >Stuart J.</Isim>
</Yazar>
<Yazar rdf:ID="Yazar_02">
  <Isim rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >Peter</Isim>
  <Email rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >norvig@gmail.com</Email>
  <Soyisim rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >Norvig</Soyisim>
</Yazar>
<Kitap rdf:ID="Kitap_01">
  <Kitap_Yazar rdf:resource="#Yazar_01"/>
  <Kitap_Yazar rdf:resource="#Yazar_02"/>
  <ISBN rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >0130803022</ISBN>
  <Kitap_Yayinevi rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >Prentice Hall</Kitap_Yayinevi>
  <Kitap_Ad rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >Artificial Intelligence: A Modern Approach</Kitap_Ad>
  <Kitap_Yil rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
  >2003</Kitap_Yil>
</Kitap>
</rdf:RDF>

```

Verilen kesitte Artificial Intelligence: A Modern Approach adlı kitap ve bu kitabın sahip olduğu yazarlar (Russel and Norvig) birer OWL class örneği olarak ontolojide yer almışlardır. Buna göre "Kitap\_01" bir Kitap individual'ıdır. Bu kitabın da yazarları Yazar\_01 ve Yazar\_02, Yazar class'ının individualleridir.

#### *Sorgulama İşlemi*

Hazırlanan sorgulama sistemi Java Platformu'nda geliştirilmiş olup JENA çatısına[8] ait uygulama geliştirme kütüphanesini kullanmaktadır. Kullanılan Java çalışma ortamı sürümü 1.5'tir ve JENA kütüphanesinin ise 2.2 sürümü kullanılmıştır.

Sistemde izlenen iş sırası öncelikle ilgili ontolojinin ağ üzerinde belli bir lokasyonda tutulan dosyasına erişimi, dosyaya eriştikten sonra ontolojinin dosyadan alınarak JENA aracılığıyla belleğe yüklenip çalışma zamanı ortamında graf modelinin oluşturulması ve bu model üzerinde RDQL kullanılarak anlamsal sorguların gerçekleştirilmesi şeklinde olmuştur.

Aşağıdaki sistem yazılım kodu parçasında ontolojinin belli bir URI'den (Universal Resource Identifier) alınarak JENA çalışma ortamında modelin oluşturulması gerçekleştirilmektedir:

```

public void load (String descriptionUrl) {
  OntModel ontModel;
  if (ontModel == null) {
    ontModel=
    ModelFactory.createOntologyModel
    (ProfileRegistry.OWL_LANG);

    ontModel.getDocumentManager().setProcessImports(false);
  }
  ontModel.read(descriptionUrl);
}

```

Ontoloji modeline ait graf oluşturulduktan sonra onun üzerinde RDQL kullanılarak sorgular işletilebilir. Sorgu işletimi için KitapSorgulama sınıfına ait metod şu şekildedir:

```

public QueryResults execQuery
(OntModel m, String queryString) {

Query query = new Query(queryString);
query.setSource(m);
QueryExecution qe =
new QueryEngine(query);
return qe.exec();
}

```

Yukarıdaki metod "queryString" parametresi ile gelen RDQL sorgusunu "m" ile temsil edilen ontoloji üzerinde işletip sonuçları JENA QueryResults koleksiyonunda döndürmektedir.

Bu koleksiyon üzerinde uygun iteratörler aracılığıyla dolaşarak RDF kaynakları çekilir ve bu kaynaklarda yer alan ontoloji elemanları (OWL Individual'lar) başka işlemler için kullanılabilirler

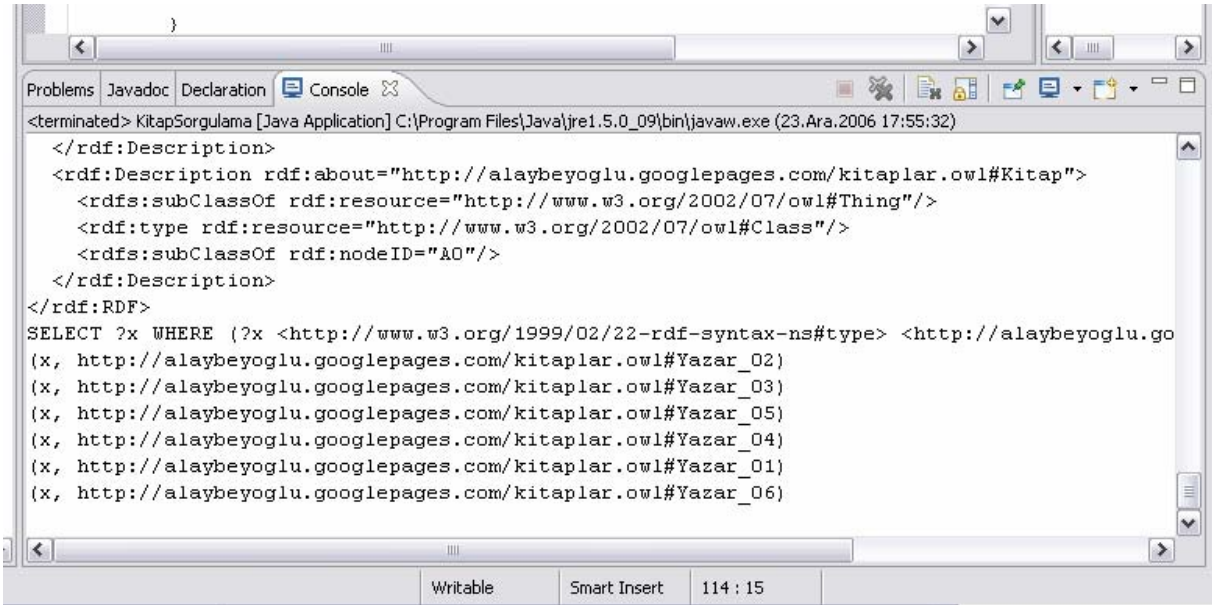
Sorgular için verilebilecek en kolay örnek ontolojideki tüm yazarların bulunduğu aşağıdaki RDQL sorgusudur:

```

SELECT ?x WHERE
(?x      http://www.w3.org/1999/02/22-rdf-
syntax-ns#type
<http://alaybeyoglu.googlepages.com/
kitaplar.owl#Yazar>);

```

RDQL sorgularının tipik bir örneği olarak WHERE tümcesi (clause) içerisinde RDF üçlüleri (RDF-triple) yer almaktadır. Üçlünün sırası subject-premise-object şeklindedir. Burada ?x ile ifade edilen sorgu sonucu dönecek QueryResults'dır. Her "x" in http://alaybeyoglu.googlepages.com/kitaplar.owl#Yazar" OWL class'ı ile http://www.w3.org/1999/02/22-rdf-syntax-ns#type ilişkisinde olduğu belirtilmiştir. Buna göre ilgili ontolojiden tipi "Yazar" olan OWL individualları bu sorgu ile istenmektedir. Şekil 3'te bu sorgunun işletilip sorgu sonuçlarının ekrana yazılması görüntülenmektedir.



Şekil 3: Ontolojideki tüm yazarların sorgulandığı RDQL sorgusunun işletilmesi sonucunda elde edilen ekran görüntüsü

Bir başka örnek de (ontolojideki sınıf ilişkileri de göz önüne alındığında) nispeten karışık olan bir yazara ait tüm kitapların sorgulanması olarak verilebilir. Aşağıdaki RDQL sorgusunda yazarının soyismi

"Daconta" olan tüm kitapların ontolojiden getirilmesi gerçekleştirilmektedir:

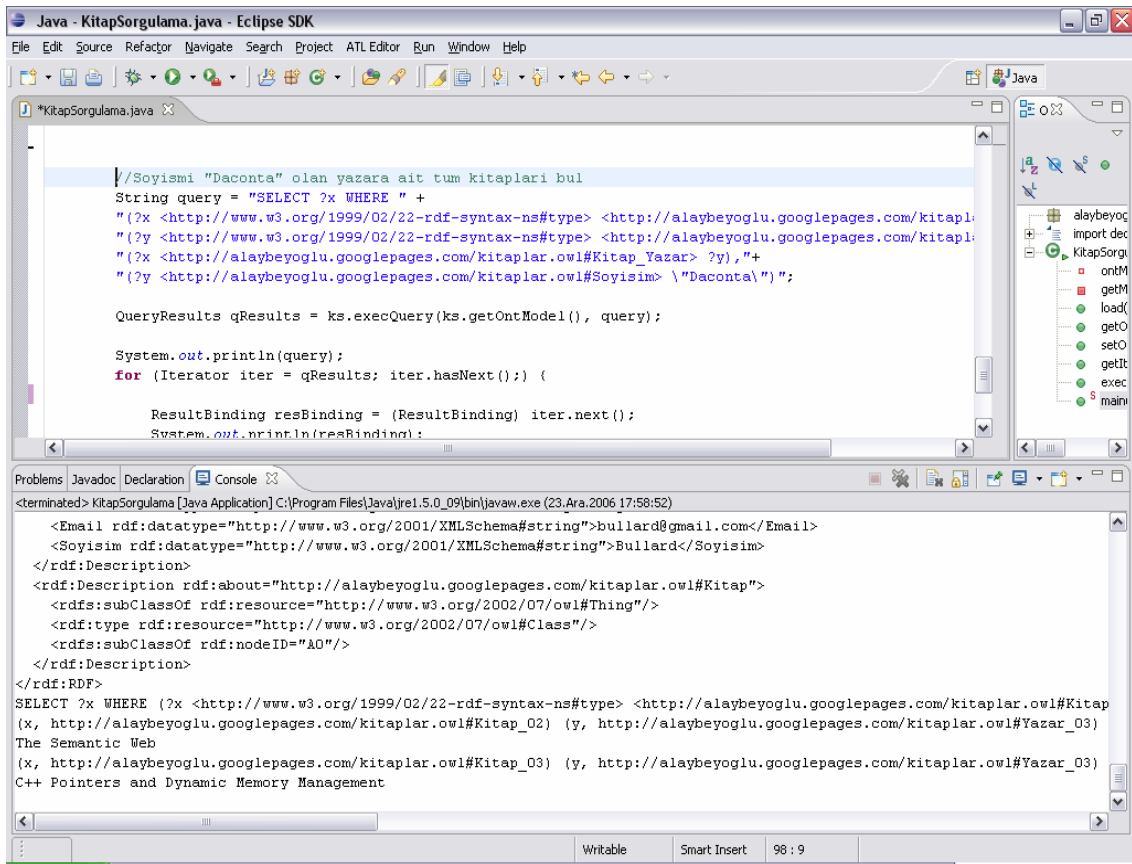
```

SELECT ?x WHERE
(?x <http://www.w3.org/1999/02/22-rdf-
syntax-ns#type
<http://alaybeyoglu.googlepages.com/
kitaplar.owl#Kitap>),
(?y <http://www.w3.org/1999/02/22-rdf-
syntax-ns#type>
<http://alaybeyoglu.googlepages.com/
kitaplar.owl#Yazar>),
(?x
<http://alaybeyoglu.googlepages.com/kit
aplar.owl#Kitap_Yazar> ?y),
(?y
<http://alaybeyoglu.googlepages.com/kit
aplar.owl#Soyisim> \"Daconta\")

```

Burada da yine sorgu sonucu istenen kaynaklar ?x ile ifade edilmektedir. İlk koşul cümlesinde ?x'in bir "Kitap" olduğu belirtilmektedir. Her bir ?y ise bir "Yazar"ı temsil eder. Her x ile y arasında (ontolojimizde yer aldığı gibi) bir "Kitap\_Yazar" ilişkisinin olacağı belirtilmiştir. Her y yazarının "Soyisim" özelliğinin aldığı değerin de "Daconta" olduğu son RDF üçlüsünde belirtilmiştir.

Bu sorguyu işleten RDQL sorgu motoru yukarıda verilen RDF üçlülerini göz önüne alarak Şekil 4'teki gibi sorgu sonuçlarını döndürmektedir:



Şekil 4: Ontolojide belli bir yazara ait tüm kitapların sorgulanması

Sorgu sonucunda da görüldüğü gibi "Daconta" soyisimli yazarın –ki kendisi OWL ontolojimizde "Yazar\_03" adlı Individual ile temsil edilmiştir- "Yazar" ilişkisinde bulunduğu iki "Kitap" kaynağı bulunmuştur. Bu kitapların "KitapAd" özellikleri ekrana yazdırılmak istendiğinde "The Semantic Web" ve "C++ Pointers and Dynamic Memory Management" değerlerinin elde edildiği görülmektedir.

#### 4. SONUÇ

Bu bildiriye, RDF tabanlı bir kitap ontolojisinin sorgulanmasına yönelik gerçekleştirilen bir çalışma anlatılmıştır. RDF tabanlı OWL dili ile ontoloji geliştirilmiş ve bu ontoloji üzerinde RDQL kullanılarak sorgu işleten ve JENA API'yi kullanan bir sistem yazılımı tasarlanıp hayata geçirilmiştir.

Ontolojinin OWL kullanılarak gerçekleştirilmiş olması bu ontoloji geliştirme dilinin W3C tarafından standart kabul edilmesi ve sıkça kullanılması açısından bize diğer uygulamalarımız için de faydalı olmuştur. JENA API ve içerisinde bulunan RDQL motoru, RDF tabanlı sorguları ontolojimiz üzerinde işletmemize olanak vermiştir. Yaptığımız sistem testlerinde sorgu çalışma zamanlarının ontoloji boyutunu (içerdiği Individual sayısı) göz önüne aldığımızda kullanıcıyı memnun edecek seviyelerde olduğu söylenebilir. RDQL oluşturmada karşılaştığımız en önemli güçlük sorgu sonucu elde edilen kaynakların özelliklerine ulaşmada oldukça karmaşık bir metod çağırımın izlenmesi zorunluluğu olmuştur. Bir diğer güçlük ise birleşim işlemi harici kesişim içeren sorguların hazırlanmasında yaşanılmıştır. Ancak söz konusu güçlüklerin RDQL yerine yeni jenerasyon olan SPARQL'un [7] vaat ettiği özelliklerle aşılabacağına inanılmaktadır. Bu çalışma dahilinde yakın zamanda planlanan ilk iş RDQL yerine SPARQL'in uygulanması ve sistem çalışmasının gözlemlenmesidir. Bir diğer hedeflenen çalışma ise Kitap ontolojisinin yeni ilişkilerle güçlendirilmesi ve içerdiği Individual sayısının artırılarak sorgu işletim performansının ölçülmesidir.

## 5. KAYNAKLAR

- [1]. T. Berners-Lee, J. Hendler, and O. Lassila, "The Semantic Web", Scientific American, vol. 184, no. 5, pp. 34--43, 2001
- [2]. Dave Beckett's Resource Description Framework (RDF) Resource Guide <http://www.ilt.bristol.ac.uk/discovery/rdf/resources>
- [3]. W3C Technical Reports and Publications <http://www.w3.org/TR>
- [4]. The Semantic Web Community Portal World Wide Website, <http://www.semanticweb.org>
- [5]. Hendler, J., et al., Spinning the Semantic Web: Bringing the World Wide Web to Its Full Potential. 2002: MIT Press.
- [6]. Ontology Web Language World Wide Website, <http://www.w3.org/TR/owl-features/>, W3C
- [7]. SPARQL, <http://www.w3.org/TR/rdf-sparql-query/>, W3C
- [8]. Jena – A Semantic Web Framework for Java, JENA, <http://jena.sourceforge.net/>, Hewlett-Packard Development Company
- [9]. Resource Description Framework, <http://www.w3.org/RDF/>, W3C
- [10]. RDF Query Language, <http://www.w3.org/Submission/RDQL/>, W3C
- [11]. Protege : Free Open-Source Ontology Editor and Knowledge-Base Framework, <http://protege.stanford.edu/>