

ANLAMSAL WEB TABANLI KÜTÜPHANE BİLGİ SİSTEMİ

Zehra Burcu KAZANDIR

**YÜKSEK LİSANS TEZİ
YÖNETİM BİLİŞİM SİSTEMLERİ**

**GAZİ ÜNİVERSİTESİ
BİLİŞİM ENSTİTÜSÜ**

HAZİRAN 2011

ANKARA

Zehra Burcu KAZANDIR tarafından hazırlanan ANLAMSAL WEB TABANLI KÜTÜPHANE BİLGİ SİSTEMİ adlı bu tez raporunun Yüksek Lisans Tez Raporu olarak uygun olduğunu onaylarım.

.....
Tez Yöneticisi

Bu çalışma, jürimiz tarafından oy birliği / oy çokluğu ile
Anabilim Dalında Yüksek lisans tezi olarak kabul edilmiştir.

Başkan : _____

Üye : _____

Üye : _____

Üye : _____

Üye : _____

Tarih :/...../.....

Bu seminer raporu, Gazi Üniversitesi Bilişim Enstitüsü seminer raporu yazım kurallarına uygundur.

TEZ BİLDİRİMİ

Tez içindeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edilerek sunulduğunu, ayrıca tez yazım kurallarına uygun olarak hazırlanan bu çalışmada orijinal olmayan her türlü kaynağa eksiksiz atıf yapıldığını bildiririm.

Zehra Burcu KAZANDIR

ANLAMSAL WEB TABANLI KÜTÜPHANE BİLGİ SİSTEMİ

(Yüksek Lisans Tezi)

Zehra Burcu KAZANDIR

GAZİ ÜNİVERSİTESİ
BİLİŞİM ENSTİTÜSÜ

HAZİRAN 2011

ÖZET

Günümüzde, İnternet ortamında yer alan kaynaklar, dinamik olmayan bir yapıda bilgiler içermektedir ve bu kaynakların/Web dokümanlarının farklı bilgisayar sistemleri tarafından verimli bir şekilde birlikte kullanılabilirliği ve paylaşılabilirliği oldukça zordur. Ancak, bilgiyi iyi tanımlanmış bir anlamla sunmak, farklı sistemlere anlamsal düzeyde bilgiyi okumayı, anlamlandırmayı, işlemeyi, tekrar-kullanmayı ve bilgi hakkında çıkarım yapmayı sağlayacaktır. Fakat güncel bilgisayar sistemleri bilgiyi sadece yazım düzeyinde işlemektedirler. Anlamsal Ağ yaklaşımı İnternetin etkinliğini çok büyük oranda arttıracak, bilginin tekrar kullanımını ve paylaşılabilirliğini sağlayarak bilgiye ek sunum gücü sağlayacaktır. Anlamsal ağ ile makinelere insanlar tarafından, anlamlandırma, çıkarsama yapma ve elverişli sonuçlar döndürerek problem çözme yetisi kazandırılacaktır. Böylece statik olmayan, metin farkındalığında, anlamsal sistemler meydana gelecektir. Bilgiler tekdüze olmuş bir yapıyla tanımlandıklarından, farklı konulardaki bilgilerin birleştirilmesi ve bu bilgilerin birlikte işlenebilirliği mümkün olacaktır. Bu tezde, Kütüphane Sorgulama Sisteminin Sematik Web Uygulaması Geliştirimi için; Semantik Web kavramları ve geleceği, bilginin Semantik Web dili ile gösterimi, ontoloji

işleme, ontoloji geliştirim ortamları ve araçları, ontolojiler üzerinden sorgulama yapma kavramları araştırılmış ve gerçekleştirilmiştir. Uygulamada, kullanıcı mevcut kitapları görebilmek için isteğine yönelik sorgu kriterlerini girerek sonuçlara erişebilir.

Bilim Kodu : 902.1.014
Anahtar Kelime : anlamsal ağ, semantik web, ontoloji, ontoloji geliştirme, ontoloji bilgi sistemi
Sayfa Adedi : 78
Tez Yöneticisi : Doç. Dr. M. Ali AKCAYOL

SEMANTIC WEB BASED LIBRARY INFORMATION SYSTEM**(M.Sc. Thesis)****Zehra Burcu KAZANDIR****GAZI UNIVERSITY
INFORMATICS INSTITUTE****MAY 2011****ABSTRACT**

At the present time, the resources which are in the Internet environment have a non-dynamic structure that consists data and different computer systems have a very hard way to use and share this resources/Web documents effectively. However, representing the data in a well-defined meaning provide reading the data in a semantic level, giving a meaning to it, processing it, reusing it and sharing it by different systems in order to inferencing. But, current computer systems process the data only in a syntax level. Semantic Web approach increases the Internet's efficiency, the data's reusability and sharability thereby the data has a more powerful representation. With the Semantic Web logic, the people give the ability of understanding, inferencing and solving problems by returning effective results to the machines. Herewith, non-static, text-aware, semantic systems will be come up. The knowledge is explained in a standardized structure thus it is possible to integrate the different location datas and give them interoperability characteristic. In this thesis, Semantic Web Application Development for Library Questioning, Semantic Web concepts and it's future, presenting data with Semantic Web language, processing ontology, ontology

development environments and tools, inference on ontologies and questioning concepts were examined and developed. In this application, the user is able to view the results with entering the questioning criterias of the related books.

Science Code : 902.1.014
Key Words : semantic web, ontology, ontology development, ontology questioning system
Page Number : 78
Tez Yöneticisi : Assoc. Prof. Dr. M. Ali AKCAYOL

TEŐEKKÜR

Çalıőmalarım boyunca deęerli yardım ve katkılarıyla beni yönlendiren Hocam Doç. Dr. M. Ali AKCAYOL'a, manevi destekleriyle beni yalnız bırakmayan Ailem'e ve attığım her adımda, her başarıda ve mutlulukta sonsuz zaman boyunca elimden tutarak beni onurlandıran Eőim, Sonsuz Aőkim Cemil Kanat SARISOY'a teőekkürlerimi ve sevgilerimi iletmekten mutluluk duyarım.

İÇİNDEKİLER

	Sayfa
ÖZET.....	i
ABSTRACT.....	iii
TEŞEKKÜR.....	v
İÇİNDEKİLER.....	vi
ŞEKİLLERİN LİSTESİ.....	viii
ÇİZELGELERİN LİSTESİ.....	x
1. GİRİŞ.....	1
2. WEB TARİHÇE.....	3
2.1. Web 1.0.....	3
2.2. Web 2.0.....	4
2.2.1. Web 2.0 özellikleri.....	4
2.2.2. Web 2.0 yöntemleri.....	5
2.2.3. Web 2.0 veri yönetimi.....	6
2.2.4. Web 2.0 veri elde edimi.....	6
2.3. Web 3.0.....	7
3. SEMANTİK WEB.....	10
3.1. Semantik Web'in Yapısı ve Bileşenleri.....	10
3.2. Semantik Web Uygulama Alanları.....	13
3.3. Semantik Web ve Yapay Zeka.....	14
3.4. Semantik Web Uygulama Modeli.....	15
3.5. Semantik Web Uygulama Geliştirme.....	16
3.6. Semantik Web Son-Kullanıcı Uygulamaları.....	17
3.6.1. Son-kullanıcı uygulamaları: Piggy Bank.....	17
3.7. Semantik Web Araçları.....	20
3.8. Mevcut Semantik Editörler ve Arama Motorları.....	21
3.9. Semantik Web Dilleri ve Araçları.....	22
3.9.1. Kronolojik sırayla semantik web dilleri.....	23
3.9.2. OWL alt dilleri yapısı.....	27
4. ONTOLOJİ.....	30
4.1. Ontoloji ve Bilgi Tabanı.....	31
4.2. Ontolojilerin Kullanım Alanları.....	32
4.3. Ontoloji Tanımlamak ve Geliştirmek.....	33
4.4. Ontoloji Yapısı.....	33
4.4.1. Namespaces.....	33
4.4.2. Başlık bilgileri (Headers).....	34
4.4.3. Veri birleştirme (Data Aggregation).....	35
4.5. Ontoloji Temel Elemanları.....	35
4.6. Ontoloji Sorgulama Dilleri.....	47
5. UYGULAMA YÖNTEMİ.....	52
5.1. Semantik Web Uygulama Mimarisi.....	52
5.2. Ontoloji Oluşturma.....	54
5.3. Oluşturulan Ontoloji Üzerinde İşlemler.....	56
5.3.1. SemWeb kütüphanesi.....	56

5.3.2. Friend of a friend (FOAF)	58
5.3.3. MemoryStore sınıfı	60
5.3.4. XmlDocument sınıfı	60
5.3.5. RdfXmlReader sınıfı	61
5.3.6. System.Linq isim uzayı	62
5.4. SemanticWeb2011 Yazılımı Arayüzleri	63
6. SONUÇLAR	70
KAYNAKLAR	73

ŞEKİLLERİN LİSTESİ

Şekil	Sayfa
Şekil 2.1. Bir web 1.0 örneği.....	3
Şekil 2.2. Web 2.0 şemasal gösterimi	7
Şekil 2.3. Web 3.0 şemasal gösterimi	8
Şekil 2.4. Web 1.0, Web 2.0, Web 3.0.....	9
Şekil 2.5. Kronolojik web	9
Şekil 3.1. Semantik web katmanları.....	12
Şekil 3.2. Semantik web uygulama modeli	16
Şekil 3.3. Semantik web uygulama geliştirme	17
Şekil 3.4. Piggy bank arayüzü.....	18
Şekil 3.5. Piggy bank arayüzü-2	19
Şekil 3.6. Last fm arayüzü.....	20
Şekil 3.7. Semantik web dilleri	23
Şekil 3.8. SWRL örnek	27
Şekil 4.1. Protégé ontoloji geliştirim aracı arayüzü	45
Şekil 4.2. Protégé ontoloji geliştirim aracı arayüzü-2.....	46
Şekil 4.3. Ontoloji Sorgulama Örneği.....	50
Şekil 4.4. Sesame veri deposu.....	51
Şekil 5.1. Semantik web uygulaması katmanlı mimari.....	52
Şekil 5.2. “Kitap.xml” dosyası altyapısı	55
Şekil 5.3. “Project.dbml” dosyası	62
Şekil 5.4. Giriş ekranı.....	63
Şekil 5.5. Kitap türleri.....	64
Şekil 5.6. Kriter girişi.....	64
Şekil 5.7. Sorgu sonucu örneği	65
Şekil 5.8. Kitap ayrıntıları ekranı	66
Şekil 5.9. Sorgu sonucu olumlu değil	67
Şekil 5.10. Sorgu sonucu olumlu	67
Şekil 5.11. Semantik kelime tamamlama örneği - 1	68
Şekil 5.12. Semantik kelime tamamlama örneği - 2	68

Şekil 5.13. Semantik kelime tamamlama örneđi - 3 69

ÇİZELGELERİN LİSTESİ

Çizelge	Sayfa
Çizelge 5.1. Sistemin katmanlı mimarisi.....	63

1. GİRİŞ

Günümüzdeki Web içeriğine temel oluşturan birçok 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 ortaya çıkarılmıştır. Anlamsal Web yeni ve ayrı bir web olmayıp, bilgilere iyi tanımlanmış anlamların verildiği, bilgisayarların ve insanların birlikte çalışmalarına imkân veren güncel Web’in bir eklentisidir [2]. Anlamsal Web kavramının ortaya çıkmasındaki ana amaç iyi tanımlanmış ve ilişkilendirilmiş olan bilgilerin ve servislerin Web ortamında farklı bilgisayar sistemleri tarafından kolay bir şekilde bilgisayarca-okunabilir ve bilgisayarca anlaşılabilir olmasını sağlayacak standartların ve teknolojilerin geliştirilmesidir ve bu amaçla birlikte kaynakların tekrar kullanılabilirlikleri ile paylaşılabilirlikleri artacaktır [3].

Güncel arama motoru teknolojileri, tüm dünya üzerinde yayımlanan Web sayfalarında arama yapma işleminde çok iyidir ancak eksik olduğu nokta yaptığı aramanın talep edilen hassaslıkta ve geri çağırırında olmamasıdır [4]. Bu nedenle önceki çalışmalarda yapılan aramalarda iki önemli kıyaslama kriteri ortaya çıkmaktadır; sorgu zamanı ve sorgu sonuçlarının kalitesi. Semantik arama ile uygulamalar, RDF (Resource Description Framework – Kaynak Tanımlama Çerçevesi) altyapısını içeren, elemanların içeride birbirleriyle ilişkilendirilerek etiketlendirildiği XML (Extensible Markup Language – Genişletilebilir İşaretleme Dili) dokümanlarının Web’de yayınlanması ve bu dosyalar üzerinde hızlı ve mantıklı sorguların yapılması yönünde gelişmektedir [5]. Semantik arama motoru gelişmiş örneklerinden Hakia, Cognition, Deepdyve, Powerset gibi teknolojiler anlamsal aramanın getirilerini çok farklı ve geniş multimedya alanında ortaya koymaktadır [6]. Bizim çalışmamızın getirisi ise özelleştirilmiş tek bir alan üzerinde işlev sunması

yönüyle kesin ve gerekli sonucu ilgili kullanıcıya ulaştırımda garantisini söz konusudur.

Bu tezde, Kütüphane Sorgulama Sistemi için Semantik Web Uygulaması Geliştirimi gerçekleştirmek amacıyla; bilginin Semantik Web dili ile sunulması, ontoloji işleme, ontolojiler üzerinden sorgulama kavramları gerçekleştirilmiştir. Bu gerçekleştirimin anlatımında öncelikle Web'in Semantik Web bir diğer deyişle Web 3.0'a kadar uzanan yol haritasından bahsedilerek uygulamamızın temelini oluşturan anlamsal Web, ontoloji, ontoloji dilleri ve ilgili teknolojiler detaylandırılarak tezimizin gerçekleştirimi hakkında geniş çapta bilgilendirilme yapılmıştır. Semantik Web uygulaması geliştiriminde aşılacak adımlar; RDF altyapısı ile ontoloji dosyası tasarımı, bu yapıda bilgi sorgulama işlemini gerçekleştirme, sorgu sonuçlarının veritabanı ile iletişimini sağlama ve farklı mimari katmanlarında olacak şekilde aynı sistem üzerinde son kullanıcıya ekranları sunma ana başlıkları halinde anlatılmaktadır.

Bu tez, altı bölümden meydana gelmiştir. Yukarıda açıklanan giriş bölümünü takiben ikinci bölümde Web Tarihçesi, Web 1.0, Web 2.0 ve Web 3.0'ın özellikleri ile birlikte anlatılmaktadır.

Üçüncü bölümde, Semantik Web yapısı, bileşenleri, uygulama alanları, son-kullanıcı uygulama örnekleri ve Semantik Web dilleri ve araçları anlatılmaktadır.

Dördüncü bölümde, ontoloji kavramı, kullanım alanları, yapısı, sorgulama dilleri ile birlikte ayrıntılı işlenmektedir.

Beşinci bölümde, tezimizin tasarım ve uygulama yöntemi aktarıldığı gibi arayüzler üzerinden yazılım işleme mantığı açıklanmaktadır.

Altıncı bölüm olan sonuçlar bölümünse ise tezimizi oluştururken karşılaştığımız durumlar, kazandığımız edinimler ve ileriki çalışmalar anlatılmaktadır.

2. WEB TARİHÇE

Kronolojik olarak web tarihçesi, Web 1.0, Web 2.0 ve Web 3.0 şeklinde geliştirilen bir yapı ve teknolojidir.

2.1. Web 1.0

Web 1.0, statik içerikli sayfalardan, animasyonlu imgelerden, çerçeve kümelerinden(frameset) ve misafir defterlerinden oluşmaktadır. Şekil 2.1’de gösterildiği gibi, Web 1.0 kullanıcıların tek taraflı olarak sahip oldukları kaynakları/bilgileri, Web ortamında başkaları tarafından değiştirilebilir özelliği olmaksızın paylaşmaları anlamına gelmektedir. Bu örneğin Web 1.0 örneği olduğunu gösterimi yapılan sayfanın değiştirilebilir özelliği olmayıp sadece görsel paylaşım sunma özelliğine sahip olmasından dolayı anlaşılmaktadır.



Şekil 2.1. Bir web 1.0 örneği

2.2. Web 2.0

Web 2.0, Syntax seviyesinde olan, sadece insan aklına ve kullanımına yönelik, metin-bilinçli bir Web yapısıdır. Web 2.0'da istenilen bilgiye ulaşmada zaman ve emek tüketimi oldukça fazla olduğu gibi, dokümanların tutarlılıklarına yönelik insan bazlı yapılan bakım maliyeti de çok yüksektir.

HTML (HyperText Markup Language – Üstünyazı İşaretleme Dili), PHP (Hypertext Preprocessor – Üstünyazı Önilemcisi), ASP (Active Server Pages - Etkin Sunucu Sayfaları), JSP (Java Server Pages – Java Sunucu Sayfaları), Flash gibi Web 2.0 Web dilleri, göstermekte oldukları verinin ne olduğundan habersizlerdir, sadece sayfaların yönetimsel mantığını ve bilginin kullanıcıya görsel aktarımını sağlayabilmektedirler.

2.2.1. Web 2.0 özellikleri

Katılım: Kullanıcı katılımı ile insan merkezli, uygulamalara dahil olma ve fikir beyan etmeye dayalı, Web yapısı hakimdir. Web Günceleri (Bloglar), Sosyal Ağlar, Görsel (Resim, Video) Paylaşım Siteleri ile örneklenmektedir. Bu özellik ile kolay içerik yaratımı ve paylaşımı sağlanmaktadır.

Standartlar: W3C Consortium, içeriğe ve uygulamalara erişimde ortak arayüzlerin kullanımı, veri işleme ve transfer etmede ortak yöntemlerin kullanımı nitelikleriyle standardize yapı sağlar. Bu özellik ile modülerlik, kararlılık ve kolay entegrasyon sağlanmaktadır.

Dağıtıklık: Uygulamaların mimarisi dağıtık tasarlanmıştır. İçeriğin, servislerin ve uygulamaların pek çok farklı bilgisayardan ve/veya sistemden kullanılabilir olması özelliğindedir. Bu özellik ile kolay kullanıcı katılımı ve kullanımı, veri güvenliği sağlanmaktadır.

Açıklık: Geliştiricilerin ve şirketlerin uygulamalarına serbest erişim sağlaması ile içeriğe transparan erişim mümkündür. Bu özellik ile paylaşımın artması, fikirlerin buluşması sağlanmaktadır.

Modülerlik: Pek çok farklı uygulamanın veya modülün sağlanan standartlar üzerinden beraber çalışmasını sağlamak, uygulama parçalarından daha güçlü bir

bütün yaratmak amaçlanmıştır. Bu özellik ile verimli altyapı, kolay birleşim sağlanmaktadır.

Kullanıcı Kontrolü: Sunulan içeriğin (veya yine kullanıcı tarafından yaratılan içeriğin) kullanıcılar tarafından kontrol edilmesi, değiştirilmesi, kullanıcıların Web'deki kendi aktivitelerinden doğan bilgiyi yönetebilmeleri mümkün kılınmaktadır. Bu özellik ile fikir beyanının sağlanması, insana özgü olanın veya ilgi odağının kendiliğinden filtrelenmesi sağlanmaktadır.

2.2.2. Web 2.0 yöntemleri

Bilgi Kümeleme: Her çeşit bilgidен merkezi yığınların oluşturulması, bunların tek bir uygulamanın veya arayüzün altında derlenmesi sağlanmaktadır.

Modül Gömme: Bir içeriği veya uygulamayı, çıktısının biçemi korunacak şekilde başka bir sayfaya birleştirmek sağlanmaktadır.

İçerik Karıştırma: Bilgi yığınlarından çıkartılan içeriklerin derlenerek yeni bir içerik yaratmada kullanılması sağlanmaktadır.

Yeni Uygulama: Farklı kaynaklardan gelen, farklı türdeki içeriklerin veya bilginin birleştirilmesi ve kombinasyonu ile yeni bir servis yaratılması sağlanmaktadır.

Etiketleme: Bir bilgi veya içeriğe onu açıklayan ve ileride gruplanmasında kullanılabilecek kelimelerin atanması sağlanmaktadır.

Folksonomiler: İnsanlar tarafından yaratılan ortak bilginin, yine insanlar tarafından, işaretleme ve diğer yöntemler ile zengin biçimde gruplandırılması sağlanmaktadır.

Etiket Bulutları: Bir konu ile ilgili bütün etiketlerin grafiksel bir gösterim içinde, önemli olanlarının daha farklı ifade edilmesi sağlanmaktadır.

Pencere Araçları (Widget): Farklı ortamlara veya Web sayfalarına entegre edilebilen küçük ve modüler web uygulamaları sağlamaktadır.

Teknolojiler: XML (eXtensible Markup Language), AJAX (Asynchronous JavaScript and XML), Python/Djang, Flex/Ai, Ruby on Rail, RSS (Really Simple Syndication) gibi örneklere sahiptir.

2.2.3. Web 2.0 veri yönetimi

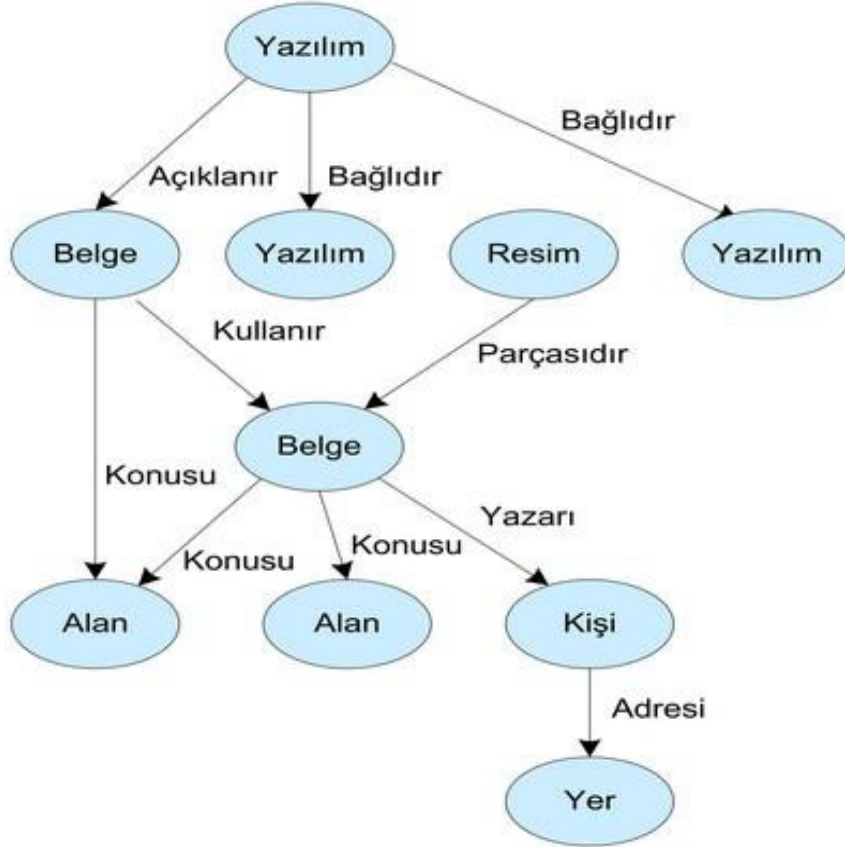
Bilgiler sunucularda tutulan büyük veritabanları içinde saklanırlar ve Web dokümanlarına yapılan sorgular, sunucular üzerinde koşan programlar tarafından koşturularak sadece “insan anlamlandırmasına” yönelik kalıcılığı olmayan çevrim içi dokümanları görsel olarak geri döndürürler.

2.2.4. Web 2.0 veri elde edimi

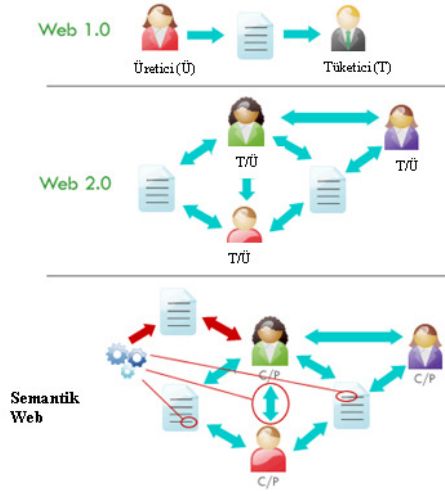
Web 2.0’da sorgu işleminin yükü tamamıyla kullanıcı üzerindedir. Kullanıcı sorgu sonucunda “bilgi” elde etmek yerine “doküman kümesi” elde eder ve bu küme içerisinde, sorgusunun cevabının olup olmadığını bilmeden analiz ve anlamlandırma sürecine girer. Şekil 2.2’de gösterildiği gibi, Web 2.0’da kaynaklar birbirlerini ve birbirleri ile olan ilişkilerini anlamlandıramamaktadırlar.

Web’den Bilgi Çekimi için Mevcut Yöntemler: Browsing(Göz Atma) ve Anahtar Kelime Tabanlı Aramadır. Browsing yöntemi ile kullanıcılar ellerinde bulunan tam URL adresleri ile bağlantılar arasında dolaşarak istedikleri kaynağa erişmeye çalışırlar. Bu yöntemin dezavantajı, kullanıcıların bağlantılar arasında kaybolma ihtimali, istedikleri kaynak Web’de bulunsa dahi ona erişememe durumları ve arama yapılırken kaybedilen zamandır. Anahtar Kelime Tabanlı Arama yönteminde ise arama motorlarında ilgili anahtar kelimenin aranması ile dönen WEB doküman kümesinde kullanıcının istediği kaynağı araştırması söz konusudur. Bu yöntemin dezavantajı, kullanıcıların ilgili anahtar kelimeyi Web’deki tam haliyle tahmin edebilme güçlüğü, aramalarına uygun arama motorları hakkında bilinçli olmaları gerekmesi ve dönen sonuçların elverişsiz sonuçlar olabilmeleridir.

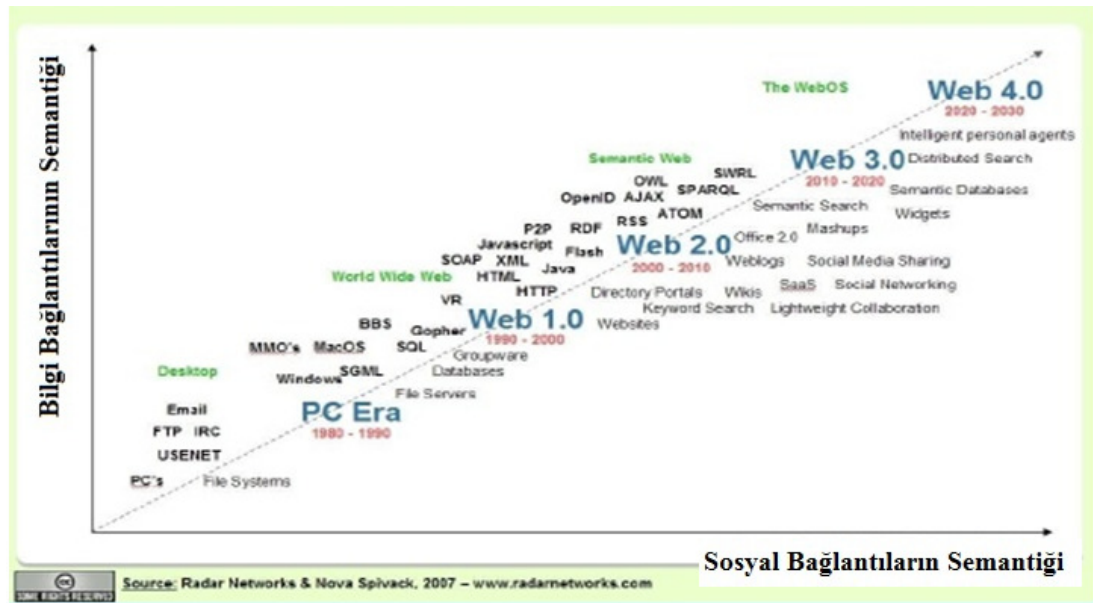
kaliteli ve verimli 3B içeriktir. Şekil 2.3'te de gösterildiği gibi Web 3.0'da kaynaklar birbirlerini tanıyabilmekte ve aralarındaki ilişkiyi anlamlandırabilmektedirler. Web 1.0, Web 2.0 ve Web 3.0 arasındaki karşılaştırmayı anlatan şekil için Bkz. Şekil 2.4 ve 2.5.



Şekil 2.3. Web 3.0 şemasal gösterimi



Şekil 2.4. Web 1.0, Web 2.0, Web 3.0 [7].



Şekil 2.5. Kronolojik web

3. SEMANTİK WEB

Semantik Web, web içeriklerinin sadece doğal dillerde değil, aynı zamanda ilgili yazılımlar tarafından anlaşılabilir, yorumlanabilir ve kullanılabilir bir biçimde ifade edilebileceği, böylece bu yazılımların veriyi kolayca bulmasını, paylaşmasını ve bilgiyi birleştirmesini sağlamayı amaçlayan gelişen bir internet eklentisidir [8]. Anlamsal web, WWW konsorsiyum (W3C)'u tarafından yürütülen ve şu anda kullanılmakta olan web mimarisini, verinin uygulamalar, kurumlar ve topluluklar arasında paylaşarak ve tekrar kullanımını artırarak, geliştirmeyi amaçlayan bir çalışmadır [9]. Anlamsal web fikrini ortaya atan ilk makale Scientific American dergisinde 2001 yılında yayınlanmıştır [10]. Bu çalışma, anlamsal web'i, şu anda kullanılan web'den ayrı bir web oluşumu değil, onun bir uzantısı olarak tanımlar. Anlamsal web'in hali hazırda kullanılan web ortamını geliştirmesinin ilk aşamasının, web ortamındaki veri ve dökümanları makinelerin de işleyip "anlayabilmesi" olacağı belirtilmiştir.

Semantik Web Teknolojisinin amacı, makinelere, insanların yaptığı gibi, Web'de gösterimi yapılan bilgiyi "okuma" ve "anlamlandırma" yetisi kazandırmaktır.

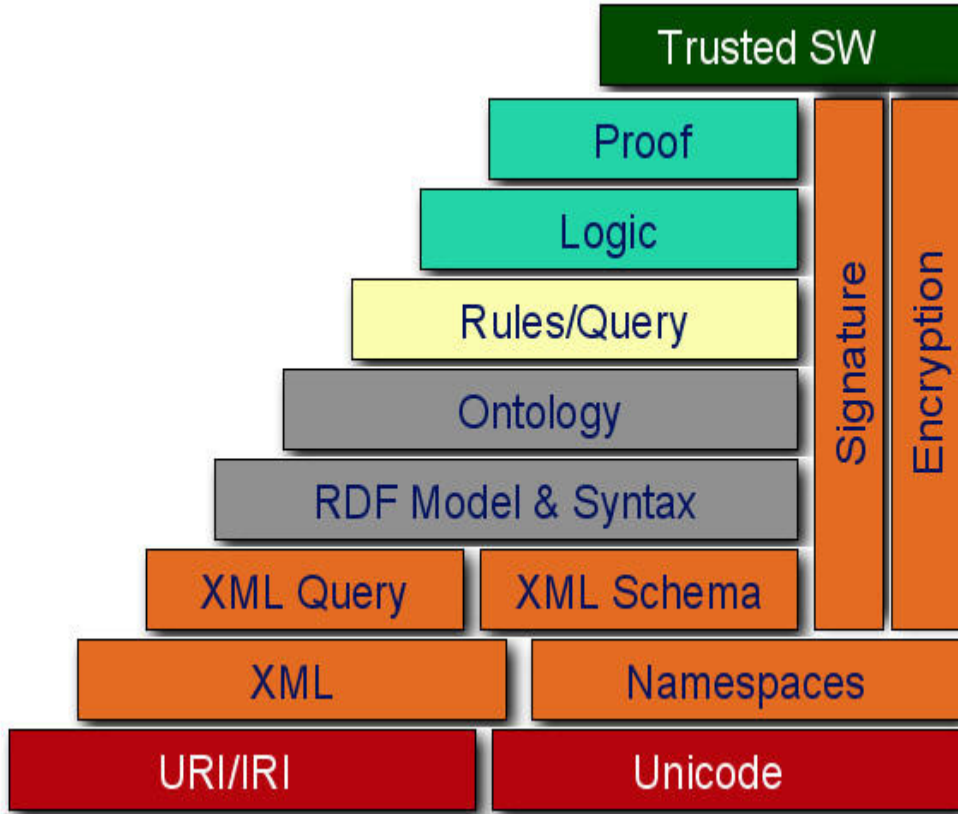
Semantik Web'in sahip olduğu anlamlı ve makineler- tarafından-anlaşılır Web kaynakları bilgisinin kullanıcıları (paylaşım ve işleme), hem "otomatize edilmiş araçlar" (arama motorları gibi) hem de "insanlar"dır. Bu kullanıcılar "ajan" olarak adlandırılır [11].

Bilginin bu şekilde farklı ajanlar tarafından kullanımı için, içerisinde belirli terim veya sınıfları barındıran standardize edilmiş, ölçülebilir, ortak anlamlı bir dil ile notasyonunun yapılmış olması gerekmektedir. Bu da *ontoloji* kavramını karşımıza çıkarmaktadır.

3.1. Semantik Web'in Yapısı ve Bileşenleri

Anlamsal ağ, yapısal olarak felsefe, tasarım prensipleri, etkileşimli çalışma grupları ve yardımcı teknolojilerden oluşur. Bu parçaların hepsinin, belirli bir problem

uzayındaki kavramları, terimleri ve bağlantıları resmi bir biçimde ifade etmesi beklenmektedir. Anlamsal ağın temelleri bir nesneyi sınıflandırmayı ve onun değerinin doğru şekilde tanımlanıp atanması gerektirir. W3C, Massachusetts Institute of Technology, Hewlett-Packard ve kullanıcı tabanındaki bazı web topluluklarınca belirlenen standartlar üzerine kurulu olan Semantik Web, sözdizimi için XML ve isimlendirme için URLs kullanan pek çok uygulamayı birleştirmek amacıyla Resource Description Framework (RDF)'den faydalanıyor. Başlangıcından bu yana, Anlamsal ağ çalışmalarına W3C çalışma grupları ve ortaya koydukları standartlar yön vermektedir. Varolan ağı veriye dönüştürmeyi amaçlayan bu çalışmalar genel olarak iki merkezde toplanmaktadır. Bir merkez, RDF (Resource Description Framework) ve OWL (Web Ontology Language) gibi dillerle anlam bilimsel cümleler kurulması ve uygulamaların bu cümleler yoluyla geliştirilmesini amaçlamaktadır. W3C nin ortaya koyduğu standartlar ışığında RDF ve OWL yaratabilmek, yaratılmış olanlar üzerinde SPARQL (SPARQL Protocol and RDF Query Language – SPARQL Protokol ve RDF Sorgulama Dili) ile sorgulama yapabilmek için geliştirilmiş pek çok araç vardır, ve pek çoğu da geliştirilmeye devam etmektedir.



Şekil 3.1. Semantik web katmanları

Şekil 3.1’de mana ağı yığını (semantic web stack) şekli ile değişik katmanlarda bulunan teknolojiler, w3c tarafından standartlaştırılmış teknolojiler gösterilmektedir.. Şekil 3.1 incelendiğinde en alt seviyede URI (Unique Resource Identifier) seviyesi görülmektedir. Bu seviye mana ağlarının bulunduğu servis sağlayıcılarının internet üzerindeki adresleridir. Yine alt seviyelerden Unicode ile anlatılmak istenen ise verinin işlenmesi ve iletimi sırasında kullanılacak olan kodlama standardıdır. Örneğin XML dosyalarının saklanması ve iletimi sırasında XML dosyalarının UTF (Unicode Transformation Format) tablolarına uygun olması istenmektedir.

Crypto katmanının en üst seviyeye kadar çıkıyor olmasının sebebi, kullanılacak olan dijital imza (digital signature) ve benzeri güvenlik uygulamalarının her katmandan kontrol edilebiliyor olması ve katmanların (örneğin XML veya RDF) güvenlik parçaları barındırmamasıdır. Örneğin XML teknolojisinin içerisinde doğrudan bir güvenlik çözümü söz konusu değildir. Yine kullanıcı katmanı (user interface & application) ile geri kalan mana ağı yığını (semantic web stack) arasında bulunan

güven (trust) katmanı da bu amaca hizmet etmektedir ve kullanıcının şifre katmanı (crypto) ile erişimini ayarlamaktadır [12].

3.2. Semantik Web Uygulama Alanları

Anlamsal web ile ilgili çalışmalar amaçları açısından temel olarak üç alanda incelenebilir;

- Bir alanla ilgili ontoloji geliştirme çalışmaları
- Geliştirilen ontolojileri işleyen araçlar ve sistemler geliştirmeyi amaçlayan çalışmalar
- Hem ontoloji geliştirme sürecini hemde ontolojik verileri işleyecek sistemleri geliştirme sürecini kolaylaştırmak için bir takım araçlar geliştirmeyi amaçlayan çalışmalar.

Günümüz anlamsal web teknolojilerinin temelini oluşturan ontolojilerin geliştirilebilmesi için çok farklı alanlarda pek çok ontoloji geliştirilmiştir. Genel amaçlı ontolojilere örnek olarak, iş modellerinin tanımlanması için ortak bir ontoloji çalışması [13] ve biomedikal araştırmalarla ilgili kullanılan materyaller, üretilen veriler ve bu veriler üzerinde yapılan analizlerle ilgili bilgileri tanımlayan ontoloji çalışması (<http://obi.sourceforge.net/index.php>) verilebilir. Bunlara ek olarak, [14], dilbilimsel verilerin tanımlanması ve paylaşımı ile bu veriler üzerinde analizler yapabilecek uzman anlamsal web uygulamalarına yardımcı olabilecek, dilbilimsel kavramlar için bir genel amaçlı ontoloji çalışmasıdır.

Yazılım ajanı tabanlı dağıtık işlem uygulamaları: Ontolojiler aracılığıyla tanımlanmış, yapılandırılmış ve anlamlandırılmış bilgiler, yazılım ajanlarının bu bilgileri taraması, harmanlaması ve kullanmasını sağlayacaktır. Bu bir çok alanda şu anda hayal edilen uygulamanın gerçekleştirilmesini sağlayacaktır.

Anlam tabanlı web arama makineleri: Ontolojiler ile tanımlanmış web kaynakları, web arama makinelerinin daha akıllı sorgulamaları yapmasına imkan verecektir.

Anlam tabanlı sayısal kütüphaneler: Anlamsal web teknolojilerinin sağladığı etkili sınıflandırma ve endeksleme yöntemleri sayısal kütüphanelerde bulunan çoklu ortam veri içeriğine ulaşımı ve sayısal kütüphaneler arası birlikte işleyebilirliği kolaylaştıracaktır.

Ontoloji destekli kurumsal bilgi yönetimi: Küresel ekonomi ile birlikte, iş gücü, sermaye, ve stok yönetimi gibi geleneksel kaynakların yanında, bilginin bir kaynak olarak kurumlarda yönetimi çok önem kazanmakta ve önemli bir üretkenlik etmeni olarak ortaya çıkmaktadır. Anlamsal web teknolojileri kurumsal bilgilerin etkin bir şekilde yönetilmesini ve kullanılmasını sağlayacaktır.

Otomatik web servisi keşfi, aktive edilmesi, karşılıklı işleyebilirliği ve izlenebilirliği: Web servisleri son zamanlarda en çok konuşulan ve web ortamında yeni fırsatlara yok açacak bir teknolojidir. Anlamsal web bu servislerin otomatik olarak bulunması, seçilmesi, çalıştırılması, karşılıklı işleyebilirliğini ve izlenmesini sağlayacaktır [15].

Örneğin UB/SPSC2 (Universal Standard Products and Services) ve UCEC3 gibi ürün ve hizmetlere uluslararası standartlar getiren ve bunların niteliklerini tanımlayan organizasyonlar, aslında yatay ontolojilerdir ve B2B süreçlerinde önemli konumları vardır. RosettaNet gibi tüm endüstriye açık, e-iş için gerekli standartları oluşturulduğu ve iş süreçleri için ortak bir dilin sağlandığı birliktelikler, bilgi teknolojileri alanında elektronik bileşenlerin, yarı iletkenlerin tanımlandığı dikey ontolojilerdir. Arabuluculuk, e-ticaretin önemli işlevlerinden birisidir. Hızla gelişen elektronik pazaryeri, alıcı ve satıcıları buluşturan sanal ortamlardır ve dinamik bir ekonomik değer değişim sistemini desteklemektedir. Gelişmiş arabuluculuk hizmetleri verebilmek için zengin ve esnek bir üst veri bilgisine sahip olmak gereklidir ve RDF gibi anlamsal web ile ilişkili teknolojileri kullanarak çeşitli modellemelerin yapılabileceği anlaşılmaktadır.

3.3. Semantik Web ve Yapay Zeka

Semantik Web'de Yapay Zeka'da yapıldığı gibi, insan dilini ve kelimelerini tamamen anlayıp anlamlandırmayı kapsamaz. Bunun yerine, insanların ekstra efor sarf ederek, var olan *tanımlanmış* veri üzerinde, *tanımlanmış* uygulamalar gerçekleştirerek, *tanımlanmış* bir problemi bir makineye çözdürme ve bu yolla makineleri veri işleyebilecek hale getirme sürecini anlatır.

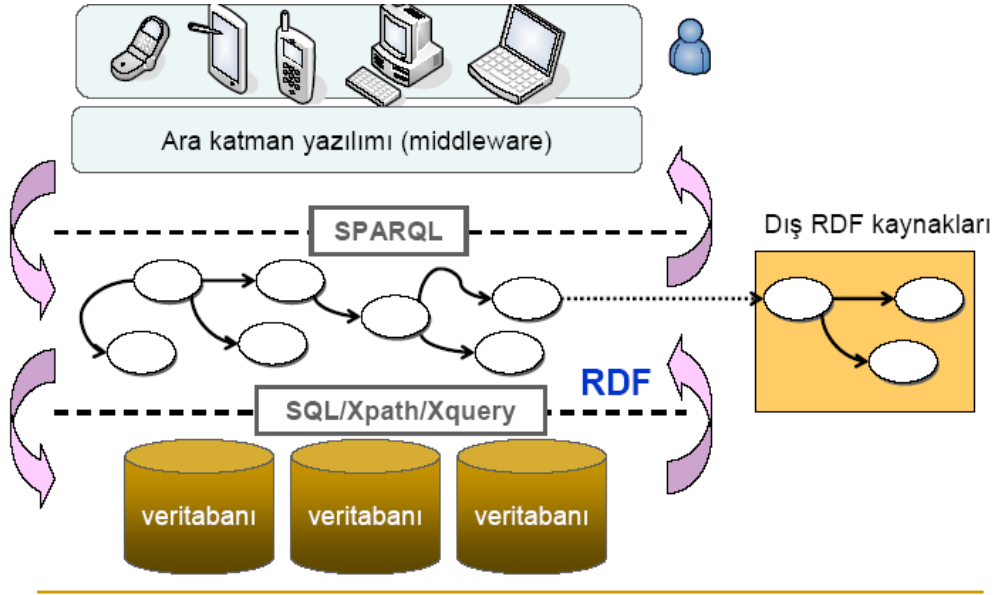
Semantic Web bir çeşit Weak AI dır. Makinalar tarafından anlaşılabilir doküman kavramı, makinaların insanlar tarafından belirlenmiş kavramları anlayabilen bir yapay zekâyı öngörmez. Fakat tam olarak belirlenmiş bir problemin, tam olarak belirlenmiş süreçlerle, var olan ve tam olarak belirlenmiş veriler üzerinde çözebilen bir makina yeteneğini kasteder. Bu bağlamda Semantic Web, büyük ölçekli bir uzman sistem gibi düşünülebilir. RDF kolay kullanılabilir olmasına rağmen, paradokları ve gereksiz söz tekrarlarını ifade edebilecek tam bir dil olacaktır. Her mekanik RDF uygulaması, RDF i bilinçli olarak kısıtlanmış bir dile çeviren, RDF kullanımını sınırlandıran ve biçimlendiren bir şema kullanacaktır. Bununla beraber, bağlantılar sadece "RDF fiilleri" arasında yapıyor olsa da, ortaya çıkan ifade büyük miktarda bilgi olacaktır [16].

Web Servisleri

Web servisleri; İnternet, intranet ve extranet üzerinde XML ve standart web protokollarını kullanarak uygulama birlikteliğini sağlayan, bilgiye erişimi kolaylaştıran, tanımlayan ve bilgiyi ortaya çıkaran yazılım uygulamalarıdır. Web servisleri, uygulamalar arasında entegrasyonu ve birlikteliği sağlayarak, iş yapmayı kolaylaştıran bir yapı sunmaktadır. Örneğin, otel bulma web servisi, uçak bileti aracılığı, araba kiralama web servisi. Web servisleri, iş süreçlerinin ve yazılımların entegrasyonunu sağlarken, grafik kullanıcı arayüzünden ve gösterimden tamamen bağımsızdır. Farklı bilgisayar sistemleri arasında veri alış verişini kolaylaştıran web servisleri, yazılım ürünleri için standartlar geliştirmekte ve firmalar arası ticaret (B2B), sipariş, sigorta kontrolleri, finansal bilgi paylaşımı ve tedarik zinciri yönetim sistemlerinde işletmenin sınır tanımaksızın gerçek zamanlı işlem yapmasına olanak sağlamaktadır. Önümüzdeki yıllarda çeşitli ilgi alanlarında kullanılacak olan web servisleri, Anlamsal Web ile uyumlu hale gelerek, gerekli yapısal entegrasyonu sağlayacaktır. Böylelikle web servisleri, Anlamsal Web vizyonunun ilk önemli uygulama alanı haline gelecektir [17].

3.4. Semantik Web Uygulama Modeli

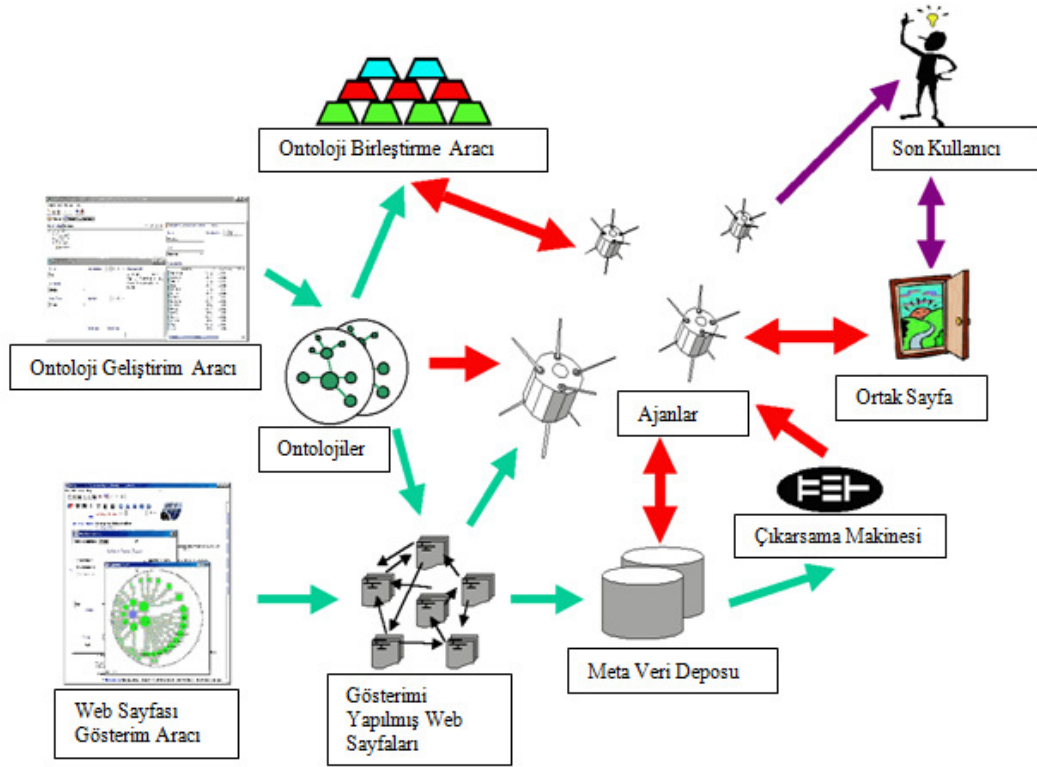
Sematik Web uygulama modeli Şekil 3.2'de gösterilmekte olduğu gibi katmanlı bir model yapısına sahiptir.



Şekil 3.2. Semantik web uygulama modeli [18]

3.5. Semantik Web Uygulama Geliştirme

Semantik Web uygulama geliştirme aşamaları Şekil 3.3'te de gösterildiği gibi ontoloji geliştirim aracından son-kullanıcıya kadar olan adımlardan meydana gelmektedir.



Şekil 3.3. Semantik web uygulama geliştirme

3.6. Semantik Web Son-Kullanıcı Uygulamaları

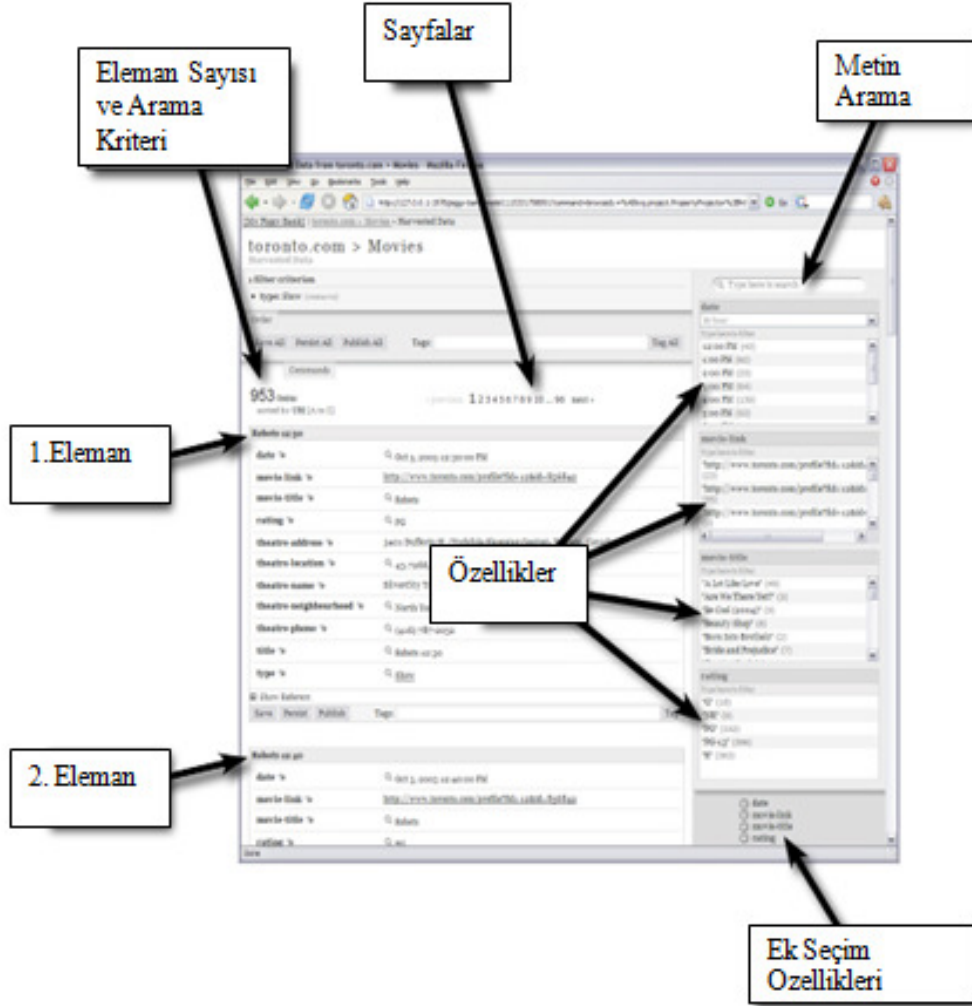
Bu uygulamaların bazıları, RDF Calender, Adobe XMP, Photostuff, SMORE, Piggy Bank, Haystack, FOAF, MusicBrainz / AudioScrobbler, Hera'dır [19-28].

3.6.1. Son-kullanıcı uygulamaları: Piggy Bank

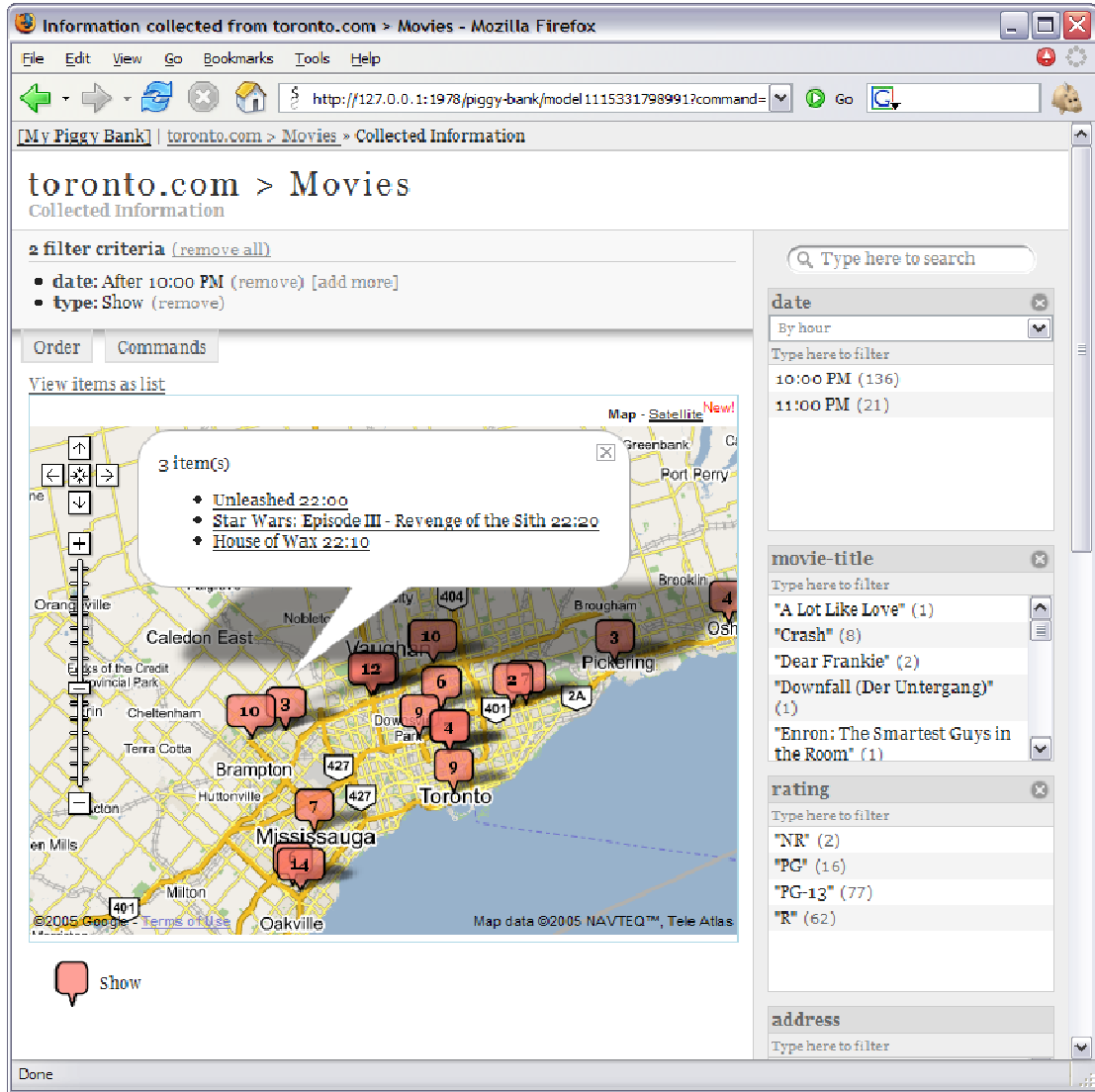
Piggy Bank, bir FireFox eklentisidir, Web browsing içine Semantik Web'i getirir. Şekil 3.4'te de gösterilmekte olduğu gibi, sorgu kriterine göre dönen cevapları ekranda göstermektedir ve ilk elemanın ayrıntılarını sunmaktadır. Şekil 3.5'te ise tıklanan cevaba göre harita üzerinde sonuç işlenmektedir. Aşamaları farklı adımlardan oluşur:

- Veriyi toplar,
- Arama ve Browse işlemi yapar,
- Bir haritada, noktaları belirler,

- Bilgiyi etiketler,
- Veri birleşimi yapar ve veriyi paylaşır.



Şekil 3.4. Piggy bank arayüzü



Şekil 3.5. Piggy bank arayüzü-2

Şekil 3.6'da gösterilmekte olan Last Fm arayüz örneğinde girilen arama kriterine göre sorguyla tam eşleşen sonuç döndüğü gibi sorguya en yakın diğer sonuçlar da gösterilmektedir.

The screenshot shows the Last.fm website interface. The browser address bar displays the URL: <http://www.last.fm/explore/?artistname=prodigy>. The page features a red header with the Last.fm logo and navigation links: Users, Music, Radio, Groups, Charts, and Help. There are also links for about, forums, downloads, and upgrade account. A login form is visible with fields for Username and Password, and a Login button. Below the header, the main content area is titled "Music on Last.fm". On the left, there are search sections for "Explore Artists:", "Find Album", and "Find Track", each with input fields and "Find" buttons. The main content area shows an "Exact Match" for "The Prodigy" with a profile picture and a note that 212,959 people who listened to The Prodigy also listen to. Below this is a "Similar Artists to The Prodigy" section with a horizontal bar chart showing the following data:

Rank	Artist	Count
1	The Chemical Brothers	100
2	Rage Against the Machine	57
2	Basement Jaxx	57
4	Red Hot Chili Peppers	49
5	Aphex Twin	48
6	Leftfield	44
6	Cypress Hill	44
8	Foo Fighters	43
9	DJ Shadow	40
10	Metallica	39
10	Deftones	39
12	Incubus	36
13	Fluke	34
13	Roni Size	34
15	Tenacious D	32

Şekil 3.6. Last fm arayüzü

3.7. Semantik Web Araçları

Gün geçtikçe Anlamsal Ağ için pek çok yeni araçlar geliştirilmektedir. Bunlardan bir kısmı aşağıda listelenmiştir:

- Platform Araç Açıklama
- Python 4Suite 4RDF XML ve RDF ayrıştırıcısı(parser)
- Python CWM veri işlemleri
- Python pyrple RDF/XML, N3, ve N-Triples ayrıştırıcısı(parser)
- Python pySesame Sesame wrapper
- Python pysparql SPARQL uç birimi
- Python RDFLib RDF kütüphanesi
- Python RDFSClosure
- Python rdfsxml.py RDF/XML ayrıştırıcısı(parser)

- Python Sparta RDF [29].

3.8. Mevcut Semantik Editörler ve Arama Motorları

Çıkarsama motoru Pellet (<http://darkparsia.com/pellet>) ve KAON2 (<http://kaon2.semanticweb.org>) ise OWL ontolojilerinden yeni ifadeler oluşturmaya olanak tanıyor [30]. Pellet de Açık Kaynak kodlu ve ücretsiz. Ama KAON2 sadece ticari olmayan maksatlar için ücretsiz kullanılabilir. Editörler ve çıkarsama motorları, anlamsal ağ teknolojisini kullanan bir dizi uygulamanın giriş düzeyini oluşturuyor.

Metaweb Technologies'in yarattığı Freebase açık veritabanı (www.freebase.com) yapılandırılmış bilgiyi hem insanların hem makinelerin kullanabileceği şekle dönüştürme hedefinde olan bir teknolojidir. Sahip olduğu verilerin temelini, Wikipedia ya da MusicBrainz (<http://musicbrainz.org>) gibi çok bilinen açık veritabanlarının makaleleri oluşturmaktadır. Öncelikle bu sitelerin içeriği çözümlenmekte, sonra ontolojiye eklenmektedir. Böylelikle Freebase şu anda üç milyondan fazla konuyu, 750.000'den fazla kişiyi, çok sayıda mekanı, firmayı ve filmi içeren bir yapıda bulunmaktadır. Ancak bunu yaparken, Freebase kendi ontolojisini ve SPARQL'dan farklı, özel bir sorgu dilini kullanmaktadır.

RDF'e dayanan bir açık veritabanı da Dbpedia (<http://dbpedia.org>). Makalelerini yine Wikipedia'dan alan bu veritabanında 2,5 milyon konu (çok sayıda kişi, yer, müzik albümü ve film) bulunuyor. Bu veriler SPARQL ile sorgulanabiliyor. Ayrıca bir dizi uygulama da arama bağlamında faydalı olabiliyor. Karmaşıklığından ötürü, genel kullanım için çok uygun değil.

Powerset adlı semantik arama motoru ise (www.powerset.com) iki bilgi kaynağının içeriğini ilişkilendiriyor. Wikipedia'nın ve daha yeni tarihli olarak Freebase'in zaten makineler tarafından okunabilen, semantik olarak hazırlanmış makalelerini kullanıyor. Power-set doğal dilde arama sorgularına da izin veriyor. Bu durumda, sonuçlardan yola çıkarak gerçeklere ulaşılması gerçekten ilginç. Örneğin "How old

is...” kalıbıyla bir politikacının yaşı sorulduğunda, gün ve tarih bilgisinin yanı sıra, bu sonucu içeren belgelerin bir listesi de dönüyor.

True Knowledge anlamsal arama motoru (www.trueknowledge.com) aynı sonuçları sunuyor ve dışarıdan Powerset’e çok benziyor. Ancak diğer sorgu biçimlerini de destekliyor. Örneğin, “Is Barack Obama older than Hillary Clinton?” (Barack Obama, Hillary Clintondan yaşça büyük müdür?) gibi sorular sorabiliyorsunuz. Üretici firma şu anda ikinci yılında bulunmaktadır.

“Yukarıdan aşağı” yaklaşımını benimseyenlerden biri de Open Calais. Document Viewer adlı ürün (www.opencalais.com/DocViewer) geleneksel bir web belgesinde bilgi satırlarında geçen nesnelere, olayları ve değerleri RDF biçiminde özetliyor. Firefox eklentisi Gnosis (<http://addons.mozilla.org/firefox/3999>) uygulama alanlarının bir örneğidir.

3.9. Semantik Web Dilleri ve Araçları

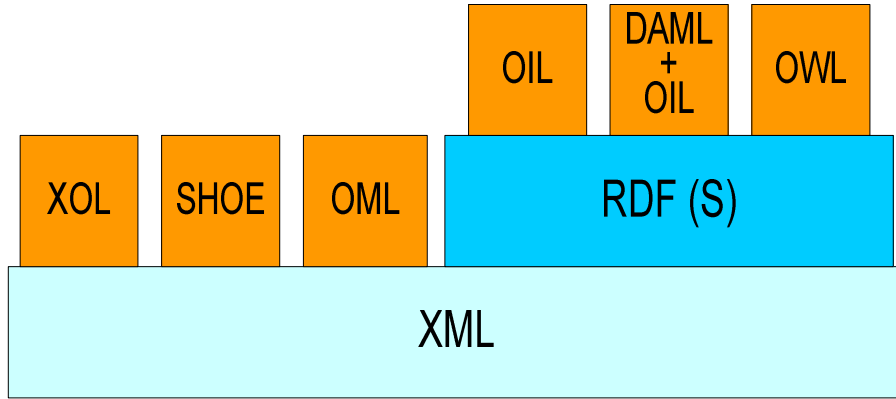
Ontolojileri tanımlamak ve çeşitlemek için kullanılan dillerdir. Bilginin içeriğini hem insanlara gösterebilip, hem de bilgisayarlar tarafından işlenebilmek üzere tasarlanmıştır.

Geçtiğimiz bir kaç yılda, XML syntax-tabanlı birtakım ontoloji dilleri geliştirilmiştir.

Bunlar;

- XOL(Ontology Exchange Language),
- SHOE(Simple HTML Ontology Extension),
- OML(Ontology Markup Language),
- RDF(Resource Description Framework)
- RDFS(RDF Schema)
- OIL(Ontology Inference Layer)
- DAML+OIL(Darpa Agent Markup Language)
- OWL(Web Ontology Language)

Bütün bu diller, Web'i organize etmek, entegre etmek, yönlendirmek ve aynı zamanda mantıksal ve ilişkisel bir tutumla içerik dokümanlarının bağlantı kurularak gruplanmasını sağlamaktadırlar.



Şekil 3.7. Semantik web dilleri

3.9.1. Kronolojik sırayla semantik web dilleri

Semantik web dilleri kronolojik olarak aşağıda listelenmektedir.

SGML (Standard Generalized Markup Language): SGML International Standards Organization(ISO) tarafından 1986 yılında kabul edilmiş ve onaylanmış bilgi yönetimi standardıdır. SGML platform-bağımsız ve uygulama-bağımsız dokümanlar yaratımını sağlamak için oluşturulmuştur. Dilbilgisi gibi bir mekanizma kullanılarak dokümanların yapısını etiketler yardımıyla tanımlamamızı sağlamaktadır.

SGML bir meta-dil dir. Yani yeni diller üretmeye müsait bir yapıya sahiptir. Burada ilginç olan nokta XML'in SGML gibi bir meta-dil den türemesine rağmen kendisinin de bir üst dil olmasıdır. Yani XML içinden de yeni diller türetilebilir [31, 32].

XML (eXtensible Markup Language): Uygulamaya yönelik işaretleme etiketleri oluşturan ve Web'de Web dokümanları ve bilgilerin yapılandırılması için evrensel bir format özelliğindeki, W3C tarafından da önerilen bir meta-dildir. Ana katkısı, Web dokümanları için ortak ve iletilebilir bir syntax sağlamasıdır. Gerek XML, gerekse XML dokümanların doğrulanması için geliştirilmiş XML Schema, bir ontoloji dili olarak düşünülmemektedir. Verileri sıralı, etiketli ağaç yapısında kodlamaktadır fakat verinin kullanımı ve anlamını belirtmemektedir. Veri deęiş-

tokuşunda XML kullanmak isteyen gruplar öncelikle söz kümelerinde, onların kullanımlarında ve anlamlarında anlaşılabilirlerdir [33, 34].

XML dört temel konuda başarı ile kullanılmaktadır [35]:

- XML uygulama bağımsız veri ve belge yaratmaktadır.
- Üst veri (meta data) ortamı için standart bir gösterim sunmaktadır.
- Veri ve belge için ortak yapısal standartlar sunmaktadır.
- XML sınanmış bir teknolojidir.

Örnek:

```
<kullanıcılar>
  <kullanıcı id="1">
    <ad>A</ad>
    <soyad>B</soyad>
  </kullanıcı>
  <kullanıcı id="5">
    <ad>E</ad>
    <soyad>F</soyad>
  </kullanıcı>
</kullanıcılar>
```

Resource Description Framework: W3C tarafından önerilen, yapısal bir üstveriyi şifrelemek, değiştirmek ve tekrar kullanmak için var olan bir doküman yapısıdır [33].

RDF, Metadayı XML formatında tanımlamak için, standart bir form oluşturur.

RDF veri modelinin üç ana nesne türü vardır:

- Kaynaklar: RDF ifadeleriyle tanımlanan her şey bir kaynaktır. Örneğin, bir HTML doküman yada Web sitesinin tamamı. (Özne)
- Özellikler: bir kaynağı tanımlamak için kullanılan bir karakteristik, nitelik ya da ilişkidir. (Yüklem)
- Cümleler: Bir kaynağın isimlendirilmiş özelliği ile ve o kaynak için o özelliğin değerinin tanımlanmış hali(Nesne), bir RDF cümlesini oluşturur.

Örnek: Tony Benn ile ilgili bir Wikipedia makalesi [36]

```
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-
ns#" xmlns:dc="http://purl.org/dc/elements/1.1/">
  <rdf:Description
rdf:about="http://en.wikipedia.org/wiki/Tony_Benn">
    <dc:title>Tony Benn</dc:title>
    <dc:publisher>Wikipedia</dc:publisher>
  </rdf:Description>
</rdf:RDF>
```

Microformat ların geliştirilmesi tek parça verinin iki farklı biçimde oluşturulup yayınlanmasının daha fazla zaman alması tarzındaki eleştiriye bir tepki olarak değerlendirilebilir. Ayrıca RDF verisinin HTML sayfalarına gömülmesini sağlayan eRDF ve RDFa gibi spesifikasyonlar mevcuttur. The GRDDL(Gleaning Resource Descriptions from Dialects of Language (Kaynak Tanımlarını Dil Lehçelerinden Ayıklama)) mekanizması varolan verinin otomatik olarak RDF gibi yorumlanabilmesini sağlar. Böylece yayıncılar sadece HTML gibi tek bir biçime ihtiyaç duyarlar [16].

RDFS(RDF Schema): RDFS ile ifade edilen veri modeli, farklı nesne-yönelimli modellemelerin(Java programlama dili gibi) kullandığı veri modeli ile çok benzerdir. RDFS için olan veri modeli, bir domain içinde, belirli verilere sahip, farklı işlevlerde ama ortak karakteristiklerde kavram daha az ya da daha çok benzer şekillerde nesne-yönelimli modeller ile benzerlik göstermektedir. RDFS (RDF Schema) gösterimi, RDF veri modelini genişleterek, alanda kullanılacak sözcük kümesini nesnelere, nesnelere arası ilişkiler, özellikler ve özelliklerin alabileceği değerler açısından tanımlamaktadır.

OIL(Ontology Inference Layer): OIL Ontoloji dili, sınıflar ve sınıf ilişkileri için tanımlamalar sağladığı gibi farklı sınıflar ve bunların özelliklerinin gösterimleri için

de limitli bir önerme sunar. İlişkiler en önemli yapıtaşını oluşturur. Sınırlı da olsa sistemleri anlamlandırmaya yönelik semantik çıkarımlar yapmaktadır.

DAML+OIL(Darpa Agent Markup Language): Ontolojileri tanımlamak için RDFS şema dilinin yeteneklerini genişleten üst seviye bir dil birleşimidir. RDFS'den üstün olmasını sağlayan özellikleri; XML Schema veri tiplerini desteklemesi, Sınıf tanımlarının diğer sınıfların ve özelliklerin terimleriyle ifade edilebilmesi şeklindedir.

OWL(Web Ontology Language): Ontoloji kelimesi felsefeden alınmış olduğu haliyle, ontoloji yani varlık bilimi, varlıklar ile bunların kategorilerinin ve birbirleriyle ilişkilerinin araştırılmasından sorumlu disiplindir. OWL ile oluşturulmuş ontoloji, sınıf tanımlamalarını, onların elemanlarını ve özelliklerini içerir.

OWL, Web'de ontolojileri yayınlamak, genişletmek ve paylaşmak için bir semantik işaretleme dilidir [38]. OWL, RDF dilinin sözlük dağarcığının genişletilmiş hali olarak ve DAML+OIL Web Ontoloji diline fazladan özellikler ekleyerek ve bazı tanımlamaları da göz ardı ederek türemiş şekildedir. DAML+OIL tasarımından öğrenilen dersler ve DAML+OIL ontoloji dilinin uygulamasını kapsayarak bir DAML+OIL düzenlemesidir. OWL bir domain içindeki sınıfların belirgin semantiğini, bu sınıflar ve elemanları arasındaki ilişkileri göstermek için kullanılır. XML, RDF ve RDFS'ten daha açıklayıcı semantik güce sahiptir.

OWL ile XML ve XML Schema dillerinin karşılaştırılması aşağıda anlatılmaktadır:

- Bir ontoloji XML Schema'dan, bir mesaj biçimi olmaması, bir veri tanımı gösterimi olması özelliğiyle ayrılır.
- OWL'in diğer bir avantajı, anlamlandırma yapacak araçların sayıca fazlaca olmasıdır. (Örneğin Racer aracı ile OWL ontolojileri anlamlandırılarak verilen cümlelerden yeni anlamlar elde edilebilmektedir.)

OWL Kod Örneği [39]:

```
<owl: Class rdf:about="http://myNS#vehicle">
  <rdfs:label=vehicle</rdfs:label>
</owl:Class>
```



```

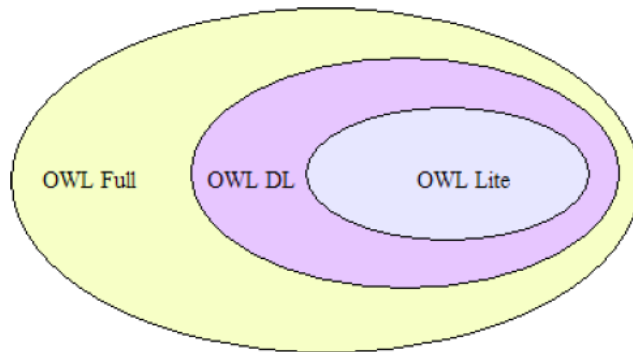
<owl:Class rdf:about="http://myNS#bicycle">
  <rdfs:label=bicycle</rdfs:label>
  <rdfs:subClassOf=<owl: Class
rdf:about=http://myNS#vehicle/=</rdfs:subClassOf>
</owl:Class>

<owl:ObjectProperty rdf:about= "http://myNS#has_pet">
  <rdfs:subPropertyOf rdf:resource=
"http://myNS#likes"/>
<rdfs:domain=<owl:Class rdf:about= "http://myNS#person"
/=</rdfs:domain>
<rdfs:rangexowl:Class rdf:about= "http:
//myNS#animal"/=</rdfs:range>

```

3.9.2. OWL alt dilleri yapısı

OWL Dilini oluşturan alt dillerin yapısı Şekil 3.8.'de gösterildiği gibidir.



Şekil 3.8. SWRL örnek [40]

OWL Lite: Basit kısıtlama özellikleri ve sınıflandırma hiyerarşilerine yönelik çalışan kullanıcılar için uygundur. “Nicelik kısıtı” özelliğini sağlar ancak sadece 0 ve 1 değerlerini kapsamaktadır. OWL dilinin tek bir kısmını kullanır. OWL DL’ nin kısıtlarını kullanır. OWL Lite sadece üst sınıf tanımlanabilir. Üst sınıflarında karar yapıları kullanılır. Sınıflar arasında eşitlik ve alt sınıf ilişkileri vardır. OWL Lite

mantıksal sonuçlarını değerlendirerek iki değer gönderir. Bu değerler 0 ve 1' dir. Özellikle durum belirteçlerinde; eğer bir çıkarım varsa 1 kabul edilir. Diğer durumlar 0 kabul edilir. Örneğin bir bilgisayarın sahip olduğu işlemci, ana kart, ram belleğine bakıldığında sunucu özelliği varsa 1 değeri gönderir, diğer durumlarda 0 değeri gönderilir.

OWL DL: Hesap yapma anlamında bütünlük eksiği olmadan en fazla anlatım gücü ve sistemleri anlamlandırmada kararlılık isteyen çalışmalara yöneliktir. Aynı zamanda bir çok kısıt içermektedir. (Örneğin tip kısıtı ile bir sınıfın aynı anda bir eleman veya bir özellik olmasını engeller.) OWL Full'un bir alt dilidir. OL ve RDF oluşturan yapılarına kısıtlar ekleyerek kullanılır. Bu kısıtlar tanımlama mantığını bağlantısını sağlar. OWL DL, terimlerin anlamlarını tanımlamada kolaylık sağlar. Otomatik hesaplamaları mümkün kılarak akıl yürütmeyi kolaylaştırır. Örneğin Ahmet Bey'in teknoloji marketinden bir adet lazer yazıcı aldığını düşünelim. Mantıksal yürütme yaparsak; eğer Ahmet Bey lazer yazıcı almışsa çok kuvvetli bir ihtimalle yazıcısı için gelecekte toner alacaktır. Mantıksal olarak lazer yazıcıya sahip bir kişinin lazer yazıcısı için toner alması gerektiği çıkarımına ulaşılarak etkin bir mantık yürütülmektedir.

OWL Full: Hesapsal anlamda garanti sunmamakla beraber, maksimum anlatım gücü ve RDF'in sözdizimsel özgürlüğü özelliklerine sahiptir. Kararlılık ve bütünlük özellikleri OWL DL'deki gibi kısıtlanmamıştır. RDF ve RDF(S) bütün bileşenlerini kullanır. RDF'in sağladığı tüm kolaylıkları sağlar. Ancak hesaplanabilirliği desteklemez. Mesela RDF de tanımlanan sınıf bir topluluğu ifade edebildiği gibi bir bireyi tanımlayan sınıf da olabilir. OWL Full daha çok tanımlama bilgisinin hesaplama bilgisinden önemli olduğu durumlarda kullanılabilir. OWL Full de bazı durumlarda otomatik mantık yürütmeler imkânsız olabilir. OWL Full dili RDF ile tam olarak uyumludur. RDF dokümanlarının yapısı aynı zamanda OWL Full dokümanlarıdır. Dezavantajlarından biri OWL Full yapıları tam kararlı değildir. Bundan dolayı kavramlar arasında tam ve etkili bir mantık yürütemez. Örneğin, Ahmet Bey bir bilgisayar malzemesi satan tekno marketten yazıcı almış olsun. Mantıksal olarak Ahmet beyin gelecekte yazıcısı için kartuş, toner veya şerit gibi yazıcı sarf malzemelerinden birini satın alabileceği olası müşteri olabileceği sonucu çıkarılabilir. Satın almış olduğu yazıcı türü belli olmadığından yazıcısı için

kullanacağı sarf malzeme çok çeşitli olabilmektedir. Bu şekilde oluşturulan mantıksal ifadede ilgi bağının zayıflığı sonucunda firmanın hangi sarf malzemeye yatırım yapması kararını verebilmesi için karşısına birden çok seçenek çıkmaktadır. Sonuç olarak bu tür mantıksal çıkarımlar tek bir sonuca ulaşmada etkin bir mantık oluşturamamaktadır.

SWRL Semantic Web Rule Language: OWL DL / Lite ve RuleML alt dillerinden tasarlanmıştır. Yazılacak kurallar *antecedent (body)* ve *consequent(head)* arasında yer alır.

SWRL Örnek:

Rule:

```

    axiom ::= rule
rule ::= 'Implies('[URIreference] { annotation }
antecedent consequent ')'
antecedent ::= 'Antecedent(' { atom } ''
consequent ::= 'Consequent(' { atom } ''

    atom ::= descripton '{' i-object ''
           | dataRange '{' d-object ''
           | individualvaluedPropertyID '{' i-
object i-object ''
           | datavalued PropertyID '{' i-object d-
object ''
           | sameAs '{' i-object i-object ''
           | differentFrom '{' i-object i-object ''
           | builtIn '{' builtinID { d-object } ''
builtinID ::= URIreference

```

4. ONTOLOJİ

Ontoloji, en öz tanımıyla, kavramsallaştırmanın bir belirtimidir [41]. Bilinen bir söz dizimi kullanılarak yazılan tanımların bir kümesini ifade eder. Belirli bir “class domain”inde bulunan konseptlerin her birinin içerdikleri çok çeşitli nitelik özelliklerinin, özellik kısıtlarının ve birbirleriyle ilişkilerinin bilinen detaylı bir tanımıdır [42].

Ontoloji, insanlar veya sistemler arasındaki iletişimin temel aldığı bir bilgi alanının, ilgili sistemler tarafından tanınan ortak anlamını ifade eder [43]. Aynı zamanda ontoloji, bilgi sistemleri alanında kullanımı gözönüne alınarak daha yalın bir tanım kullanılarak “kavramların detaylandırılması” olarak tanımlanmıştır [44]. Daha açık bir ifadeyle, ontoloji; bir ajan ya da ajan topluluğunun sahip olabileceği kavramların ve ilişkilerin tanımıdır. Nelerin ontoloji olarak kabul edilebileceği konusunda pek çok tartışmalar olmakla birlikte, tüm yaklaşımlarda temel olarak bulunan iki unsur vardır:

- Bir bilgi alanından seçilen şeyleri karşılayan terimler kümesi
- Bu terimlerin anlamlarını, belirli bir mantık çerçevesinde, detaylandırılması

Özellikle bilgi tabanı (knowledge base) ile ontoloji kavramları sıklıkla karıştırılan kavramlardır. Ontoloji ve bilgi tabanlarının ortak özellikleri; ikisinin de bilgiyi anlatmasıdır. Ancak, bilgi tabanı, ontolojiden farklı olarak, kavramlar ve bunların anlamlarından oluşan terminolojilere ek olarak bu kavramların örneklerini de içerir. Yani, eğer bilgi tabanından kavramların örnekleri çıkarılırsa, ontoloji elde edilmiş olur.

Genel anlamda ontolojiler, kavramları betimleyen terimler kümesi, bunların anlamlarının detaylı tanımları ve bu kavramlar arasında varolan ilişkilerin 4 birleşiminden oluşur. Bu kavram ve ilişkilerin yönetimini sağlayabilecek birtakım kurallar ve kısıtlarda kullanılarak, ilgili terimle ilgili olası yorum sayısı sınırlanabilir. Bunun sonucunda terimin anlamsal yapısı kısıtlanmış olacağı için, istenen bilgiye ulaşım daha kolaylaşacaktır. Bu özelliği ile ontolojiler, gerçekleştirimlerinden

bağımsız olarak farklı sistemlerin bilgi düzeyinde entegrasyonunu ve dolayısıyla bilgi paylaşımını sağlayacak altyapıyı sağlarlar. Ontolojiler, kavramları ve bunların anlamlarını öz ve belirsizlik içermeyen bilgi formunda sunabilmelidir. Ancak böyle bir bilgi, farklı ajanlar (insan ve/veya makineler) arasında paylaşılıp tekrar kullanılabilir. Ontolojiler temelde iki şekilde kullanılmaktadır. Birincisi; bir alanla ilgili kavramların sunulmasıdır. İkincisi ise; belirli alanlarla/sistemlerle ilgili geliştirilmiş olan ontolojiler arasındaki ilişkileri tanımlayarak farklı ontolojileri birbirine bağlayan ontolojilerdir. Özellikle ikinci türden ontoloji mimarileri, farklı sistemlerdeki bilgileri eşleştirmeyi/birleştirmeyi amaçlayan uygulamalar için çok önemlidirler. Bu tür uygulamalarda, formatında anlaşma sağlanmış XML DTD'leri ya da XML şemalarının kullanımı, ontolojilerdeki mantıksal yapıları sağlayamadıkları için, yapılacak işin gereken güvenilirlikte yapılmasını engellemektedir.

- Ontolojiler, bilgiyi paylaşmak isteyen herhangi bir sistem tarafından kullanılabilirler; Örneğin, veritabanı uygulama programları veya domain bağımsız uygulamalar ve yazılım sistemleri/ajanları.
- Bir ontolojide bir takım domainleri tanımlamak için “sınıf tanımları” en ortak yaklaşımdır. Örneğin, *pizza* sınıfını tanımlayan bir sınıf var olan bütün *pizza* örneklerini içinde barındırır. Sınıfların kalıtım ilişkileri kurulabildiğinden, bir sınıftan daha özel tanıma sahip sınıflar türetilebilir. Örneğin; “*acılı pizza*” ve “*acısız pizza*” örneklerinde *pizza* sınıfı bu iki sınıfın üst sınıfı(ailesi) olacaktır.

4.1. Ontoloji ve Bilgi Tabanı

Ontoloji ve bilgi tabanları karşılaştırılacak olursa aşağıdaki maddeler karşımıza çıkmaktadır.

- Bu iki kavram için Ontoloji'nin bittiği ve Knowledge Base'in başladığı sınır çizgisi belirgindir. Bir ontoloji içinde bulundurduğu sınıfın, bireysel örnekleriyle(elemanlarıyla) birlikte oluşumuyla “knowledge base” kavramını meydana getirir [42].

- Ontoloji sınıfların bireysel örnekleriyle ilgilenmemektedir. Örneğin, bir ontoloji için “*acılı pizza*” ların eleman sayısı değil o ontoloji için *pizzanın* tanımı yani bilginin tanımlanması önemlidir.

4.2. Ontolojilerin Kullanım Alanları

W3C’deki Web Ontoloji Çalışma Grubu’nun yayınladığı alanlar ontolojilerin kullanım alanları aşağıda listelenmektedir [45]:

- Kontrol edilen sözlükler
- Web sitesi ya da doküman organizasyonu ve yönlendirme desteği
- Browsing desteği
- Semantik arama yapma desteği
- Aramayı genelleştirme özelleştirme
- Anlamlandırma desteği
- Kısıtlamaların kullanımıyla uyumluluk kontrolü
- Otomatik tamamlama
- Veri ve Operasyonun birlikte işlerliği
- Doğrulama ve onaylama testleri desteği
- Konfigürasyon desteği
- Yapısal, kıyaslamalı ve özelleştirilmiş arama

Bu kullanım alanlarında uygulama teşkil etmekte olan ontoloji araçları aşağıda listelenmiştir [46]:

- Apollo (<http://apollo.open.ac.uk/>)
- LinkFactory (<http://www.landc.be/>)
- OntoEdit (http://www.ontoprise.de/products/ontoedit_en)
- OILed (<http://oiled.man.ac.uk/>)
- Ontolingua (<http://www-ksl.stanford.edu/>)
- Ontosaurus (<http://www.isi.edu/isd/ontosaurus.html>)
- OpenKnoME (<http://www.topthing.com>)
- Protege 2000 (<http://protege.stanford.edu/>)
- SymOntoX (<http://www.symontox.org>)

- WebODE (<http://webode.dia.fi.upm.es/>)
- WebOnto (<http://kmi.open.ac.uk/prjects/wenonto>)
- SESAME (<http://www.openrdf.org>)
- RDFDB (<http://guha.com/rdfdb>)
- RDFSTORE (<http://rdfstore.sourceforge.net>)
- JENA (<http://hpl.hpcom/semweb/jena-top.html>)
- KAON (<http://kaon.semanticweb.org>)

4.3. Ontoloji Tanımlamak ve Geliştirmek

Bir ontoloji pratik olarak; kavram veya varlık sınıflarının tanımlanması, sınıfların hiyerarşilerinin düzenlenmesi ve varlıklara ait örnekler ve özellikler arasındaki ilişkilerin tanımlanması ile geliştirilmektedir.

Bunlar da teknik olarak ontoloji geliştirme sürecinin, günümüz yazılım endüstrisinde kabul görmüş Nesneye Yönelik Programlama (Object Oriented Programming) ve Birleşik Modelleme Dili (UML, Unified Modelling Language) kurallarına göre yapılması gereken bir süreç olduğunu ortaya çıkarmaktadır [47].

4.4. Ontoloji Yapısı

Ontoloji yapısı, namespaces, başlık bilgileri ve veri birleştirme yapılarından oluşmaktadır.

4.4.1. Namespaces

OWL belgesi hazırlanmadan önce hangi tür kelimelerin kullanılacağı tanımlanmalıdır. Bu tanımla XML-namespace düzeninde yapılmalıdır *Namespace* tanımlamaları `rdf:RDF` etiketi ile belirtilmelidir.

`<rdf:RDF` *Namespace tanımlama alanı*

```

xmlns      = "http://www.w3.org/TR/2003/PR-owl-guide-20031209/wine#"
xmlns:vin  = "http://www.w3.org/TR/2003/PR-owl-guide-20031209/wine#"
  xmlns:base= "http://www.w3.org/TR/2003/PR-owl-guide-20031209/wine#"
  xmlns:food= "http://www.w3.org/TR/2003/PR-owl-guide-20031209/wine#"
  xmlns:owl=  "http://www.w3.org/2002/07/owl#"
  xmlns:rdf=  "http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:rdfs= http://www.w3.org/2000/01/rdf-schema#
xmlns:xsd=  "http://www.w3.org/2001/XMLSchema#"

```

4.4.2. Başlık bilgileri (Headers)

OWL dil yapısında başlık bilgileri aşağıda verilen örnekte de gösterildiği gibi `rdf:about` etiketi ile birlikte açıklanmaya başlanan bir yapıdır. Bu bilgiler içinde yorumlar, versiyon bilgisi ve diğer ontolojiler ile olan ilişkiler anlatılmıştır.

```

owl:Ontology
  rdf:about: Ontoloji hakkındaki referans veya isim
  rdfs:comment: Ontoloji hakkında açık bilginin verildiği alan
  owl:imports: Sistemde yer alan başka bir ontolojinin dahil edilmesi

<owl:Ontology rdf:about= ""> başlık bilgilerinin başladığı alan
  <rdfs:comment>An example OWL ontology</rdfs:comment>
yorumlar
  <owl:priorVersion>
    <owl:Ontology
      rdf:about= " http:
//www.w3.org/TR/2003/CR-owl-guide-20030818/wine"/>
    </owl:priorVersion>

```



```

    <owl:imports Diğer ontolojiler ile bağlantı
      rdf:resource= "http://www.w3.org/TR/2003/CR-
owl-guide-20031209/food"/>
    <rdfs:comment>Derived from the DAML Wine ontology at
http://ontolingua.stanford.edu/doc/chimera/ontologies/wine
es.daml . Substantially changed, in particular the Region
based relations.
    </rdfs:comment>
    <rdfs:label>Wine Ontology</rdfs:label> Bilgilerin
gruplandırılması/etiketlenmesi
</owl:Ontology>

```

4.4.3. Veri birleştirme (Data Aggregation)

OWL kendi yetenekleri ile birçok veri dağıtık ortamlardan toplayabilir ve çıkarsama yapabilir. İki ontolojide, mernis numarası özelliği var ise, çıkarsama ile farklı iki ontolojilerdeki kişilerin aynı olduğu sonucu çıkarabilir ve farklı bilgiler birleştirilebilir [48].

4.5. Ontoloji Temel Elemanları

Sınıf tanımlamaları

Sınıf tanımlamaları, sınıf-alt sınıf ilişkilerini anlamlandırmak için kullanılır.

rdf:id: Sınıf isimlerinin tanımlandığı alan

rdfs:subClassOf

```
<owl:Class rdf:ID= "PotableLiquid">
```

```
  <rdfs:sunClassOf rdf:resource= "#ConsumableThing" />
```

```
  ...
```

```
</owl:Class>
```

Individuals

Bir sınıfa ait üyelerin tanıtılması ve farklı şekillerde tanımlama yapılabilir.

```

<owl:Thing rdf:ID= "CentralCoastRegion" />

<owl:Thing rdf:about= "#CentralCoastRegion" >
  <rdf:type rdf:resource= "#Resource" />
</owl:Thing>

```

Özelliklerin Tanımlanması

Veri tipi özelliği veya nesne özelliği tanımlanabilmektedir. Veri tipi özelliği sınıfların örnekleri arasındaki ilişkiyi, nesne özelliği ise iki sınıfın örnekleri arasındaki ilişkiyi temsil etmektedir.

```

<owl:ObjectProperty rdf:ID= "madeFromGrape" >
  <rdfs:domain rdf:resource= "#Wine" />
  <rdfs:range rdf:resource= "#WineGrape" />
</owl:ObjectProperty>

```

```

<owl:ObjectProperty rdf:ID= "course" >
  <rdfs:domain rdf:resource= "#Meal" />
  <rdfs:range rdf:resource= "#MealCourse" />
</owl:ObjectProperty>

```

Domain & Range

Domain kavramı programlama tipler arasındaki tutarlılığı kontrol etmek için kullanılır. OWL dünyasında ise, *domain* kavramı tipleri çıkarsamak için de kullanılabilir.

Restriction

Bir şarap en az bir ŞarapÜzümünden yapılmak zorundadır:

```

<owl:Class rdf:ID= "Wine" >
<rdfs:subClassOf rdf:resource="&food;PotableLiquid" />
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#madeFromGrape"/>

```

```

        <owl:minCardinality
rdf:datatype="&xsd;nonNegativeInteger">1</owl:minCardinal
ity>
    </owl:Restriction>
</rdfs:subClassOf>
</owl:Class>

```

OWL Veri Tipleri

xsd:string	xsd:normalizedString	xsd:boolean	
xsd:decimal	xsd:float	xsd:double	
xsd:integer	xsd:nonNegativeInteger	xsd:positiveInteger	
xsd:nonPositiveInteger	xsd:negativeInteger		
xsd:long	xsd:int	xsd:short	xsd:byte
xsd:unsignedLong	xsd:unsignedInt	xsd:unsignedShort	xsd:unsignedByte
xsd:hexBinary	xsd:base64Binary		
xsd:dateTime	xsd:time	xsd:date	xsd:gYearMonth
xsd:gYear	xsd:gMonthDay	xsd:gDay	xsd:gMonth
xsd:anyURI	xsd:token	xsd:language	
xsd:NMTOKEN	xsd:Name	xsd:NCName	

Geçişli Özellik

```

<owl:ObjectProperty rdf:ID= "locatedIn" >
  <rdf:type rdf:resource="&owl;TransitiveProperty" />
  <rdfs:domain rdf:resource="&owl;Thing" />
  <rdfs:domain rdf:resource="#Region" />
</owl: ObjectProperty>

```

```

<Region rdf:ID= "SantaCruzMountainsRegion" >
  <locatedIn rdf:resource="#CaliforniaRegion" />
</Region>

```

```

<Region rdf:ID= "CaliforniaRegion" >
  <locatedIn rdf:resource="#USRegion" />
</Region>

```

Simetrik Özellik

```

<owl:ObjectProperty rdf:ID= "adjacentRegion" >
  <rdf:type rdf:resource="&owl;SymmetricProperty" />
  <rdfs:domain rdf:resource="#Region" />
  <rdfs:range rdf:resource="#Region" />
</owl: ObjectProperty>
<Region rdf:ID= "MendocinoRegion" >
  <locatedIn rdf:resource="#CaliforniaRegion" />
  <adjacentRegion rdf:resource="#SonomaRegion" />
</Region>

```

Fonksiyonel Özellik

```

<owl:ObjectProperty rdf:ID= "husband" >
  <rdf:type rdf:resource="&owl;FunctionalProperty" />
  <rdfs:domain rdf:resource="#Woman" />
  <rdfs:range rdf:resource="#Man" />
</owl: ObjectProperty>

```

Ters Fonksiyonel Özellik

```

<owl:ObjectProperty rdf:ID="has social security number" >
  <rdf:type
rdf:resource="&owl;InverseFunctionalProperty" />
  <rdfs:domain rdf:resource="#Person" />
  <rdfs:range rdf:resource="#SocialSecurityNumber" />
</owl: ObjectProperty>

```

Tersi Özelliği

```

<owl:ObjectProperty rdf:ID="hasChild" >
  <owl:inverseOf rdf:resource="#hasParent" />
</owl: ObjectProperty>

```

Bütün Değerleri, Bazı Değerleri

```

<owl:Class rdf:ID= "Wine" >
  <rdfs:subClassOf  rdf:resource="&food;PotableLiquid"
/>
  ...
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource= "#hasMaker" />
      <owl:allValuesFrom rdf:resource= "#Winery" />
    </owl: Restriction>
  </rdfs: subClassOf>
  ...
</owl:Class>

```

En Önemlilik Özelliği

```

<owl:Class rdf:ID= "Vintage" >
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource= "#hasVintageYear" />
      <owl:cardinality rdf:datatype=
"&xsd;nonNegativeInteger">1</owl:cardinality>
    </owl:Restriction>
  </rdfs: subClassOf>
</owl:Class>

```

Değeri Olma Özelliği

```

<owl:Class rdf:ID= "Burgundy" >
  ...
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource= "#hasSugar" />
      <owl:hasValue rdf:resource= "#Dry"/>
    </owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>

```

```

    </owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>

```

Ontolojilerin Eşlenmesi (Mapping)

Ontolojilerin daha fazla etkili olabilmesi için paylaşılması gerekiyor. Her paylaşımında, daha az gayret göstermek ancak tekrar kullanım (*reuse*) ile mümkün olabilir.

Eşit Sınıf, Eşit Özellik

```

<owl:Class rdf:ID= "Wine" >
  <owl:equivalentClass rdf:resource= "&vin;Wine" />
</owl:Class>

```

```

<owl:Class rdf:ID= "TexasThings" >
  <owl:equivalentClass>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#locatedIn" />
      <owl:someValuesFrom
        rdf:resource="#TexasRegion" />
    </owl:Restriction>
  </owl:equivalentClass>
</owl:Class>

```

İlişki

subClassOf

equivalentClass

Uygulamalar

TexasThings(x), locatedIn(x,y) ve TexasRegion(y)'yi içerir.

TexasThings(x), locatedIn(x,y) ve TexasRegion(y) 'yi içerir ve locatedIn(x,y) ve TexasRegion(y), TexasThings(x)'i içerir.

Birbirleriyle Aynı Bireyler Özelliği

```

<Wine rdf:ID= "MikesFavoriteWine">
  <owl:sameAs rdf:resource="#StGenevieveTexasWhite" />

```

```
</Wine>
```

OWL dünyasında, tekil isim varsayımı yoktur. Her farklı isim farklı anlamlar taşımak zorunda değildir. *sameAs* ifadesi ile sınıfların eşit olma durumu düşünülmemelidir. Bu ifade, *individual*'lar arasındaki eşitliği göstermektedir. OWL-Full dilinde ise, *sameAs* ifadesi tüm eşitlikler için kullanılabilir.

Birbirinden Farklı Bireyler Özelliği

```
<WineSugar rdf:ID= "Dry">
<WineSugar rdf:ID= "Sweet">
    <owl:differentFrom rdf:resource="#Dry" />
</WineSugar>
```

```
<WineSugar rdf:ID= "OffDry">
    <owl:differentFrom rdf:resource="#Dry" />
    <owl:differentFrom rdf:resource="#Sweet" />
</WineSugar>
```

Kesişim – OWL DL (Bazı Kullanımlar)

```
<owl:Class rdf:ID= "WhiteWine">
    <owl:intersectionOf rdf:parseType= "Collection">
    <owl:Class rdf:about= "#Wine" />
        <owl:Restriction>
            <owl:onProperty rdf:resource="#hasColor/>
            <owl:hasValue rdf:resource="#White/>
        </owl:Restriction>
    </owl: intersectionOf >
</owl:Class>
```

Birleşim – OWL DL

```
<owl:Class rdf:ID= "Fruit">
    <owl:unionOf rdf:parseType= "Collection">
    <owl:Class rdf:about= "#SweetFruit" />
```

```

    <owl:Class rdf:about= "#NonSweetFruit" />
    </owl: unionOf >
</owl:Class>

```

Bütünleyen – OWL DL

```

<owl:Class rdf:ID= "ConsumableThing">

<owl:Class rdf:ID= "NonConsumableThing">
    <owl:complementOf rdf:resource= "#ConsumableThing"/>
</owl:Class>

```

Herhangi Biri – OWL DL

```

<owl:Class rdf:ID= "WineColor">
    <rdfs:sunClassOf rdf:resource= "#WineDescriptor"/>
    <owl:oneOf rdf:parseType= "Collection"/>
    <owl:Thing rdf:about= "#White"/>
    <owl:Thing rdf:about= "#Rose"/>
    <owl:thing rdf:about= "#Red"/>
    </owl:oneOf>
</owl:Class>

```

Kesişimi Olmama Özelliği – OWL DL

```

<owl:Class rdf:ID= "Pasta">
    <rdfs:sunClassOf rdf:resource= "#EdibleThing"/>
    <owl:disjointWith rdf:resource= "#Meat"/>
    <owl:disjointWith rdf:resource= "#Fowl"/>
    <owl:disjointWith rdf:resource= "#Seafood"/>
    <owl:disjointWith rdf:resource= "#Dessert"/>
    <owl:disjointWith rdf:resource= "#Fruit"/>
</owl:Class>

```


Ontoloji ve İlişkisel Veritabanları

- Ontolojiler veri saklama alanı değil, o veri için bir tanımlama modeli oluşturur. İlişkisel veritabanları ise birer veri ambarıdır.
- Ontolojilerin amacı, verilerin erişimi ve işlenmesi için bir filtre ya da framework görevi görmek iken İlişkisel Veritabanları, ontolojiler tarafından tanımlanan farklı veri örneklerini saklamaktır.
- İlişkisel Veritabanında sorgu koşturulduğunda dönen veri, belirli kısıt ve koşulları sağlamasına göre önceden veritabanına kaydedilmiş veri ile aynıdır ancak ontolojideki sorgu sonucu geri dönen veri, belli bir anlamlandırma sürecinden geçen, daha önceden saklanıyor olmayan ve ontolojinin belirttiği bilgilere göre “üretilmiş” bir veridir.
- Ontolojide özel birtakım sorgular “ilişkiler” için sorgu yapılabilir, sıradan İlişkisel Veritabanlarında bu mümkün değildir.

Ontoloji vs. Nesne-Yönelimli Modelleme

- Özelliklerin tanımlanması; Ontoloji için “özellikler” en önemli kavramlardan biriyken, Nesne Yönelimli dünyada, özellikler sınıf tanımlamalarına dahil olan bir kavramdır.
- Ontolojide çoklu kalıtım tanımlanabilirken, Nesne Yönelimli modellerde bu mümkün değildir. Farklı üst sınıflarda tanımlı metot imzalarının çakışmaması için sınıflar arasında tekli kalıtım uygulanabilir.
- Ontolojiler, özelliklerden kalıtım yapılmasına izin vermektedir ancak nesne-yönelimli modellemede bu söz konusu değildir. Ontolojiler, farklı sınıflar arasında kullanıcı tanımlı ilişkilerin olmasına izin verirken, nesne-yönelimli modelleme sınıf-üst sınıf arasındaki ilişkiyi kısıtlamaktadır. Ancak, nesne-yönelimli modellemenin ve UML (Unified Modelling Language - Birleşik Modelleme Dili)’in geniş çapta kabul görüşü ve kullanışı nedeniyle, ontolojilerin modellenmesinde pratik tanımlamalar olarak kabul edilmektedirler. Fakat, nesne-yönelimli modelleme yaklaşımının mantıksal yeteneklilik eksiğinin olması nedeniyle, bu iki kavram tamamıyla birleştirilememektedir. Güncel olarak da nesne-yönelimli modellemeye fazladan mantıksal yetenek eklemek üzere emekli çalışmalar devam etmektedir. (OCL:Object Constraint Language, bunun bir örneğidir.)

Ontoloji Öğrenme

Ontoloji tanımlamadaki en büyük problem çok fazla bilgi var ve bu bilgilerin insanlar tarafından tekrardan işlenmesi çok zor bir işlemdir. Bunun için ontolojilerin otomatik olarak tanımlanması için birçok teknik ve araç geliştirilmiştir.

Bu alanda yapılan çalışmalar Ontoloji öğrenme (Ontology learning) adı altında toplayabiliriz. Ontoloji öğrenme: var olan verilerden (veritabanı, metin ... v.s.) otomatik veya yarı otomatik olarak kavramlar/anlam çıkartma ve bu kavramlar arasındaki ilişkileri kurma işlemidir [49, 50].

Ontolojiler anlamsal ağın temelini oluşturduklarından önemli bir yere sahiptirler. Ontolojiler üzerinde birçok çalışma yapılmaktadır. Bunların en önemlilerinden biride ontoloji öğrenmedir (ontoloji learning). Ontoloji öğrenme bir veri kaynağından ontolojiler oluşturmayı hedefler.

Bazı ontoloji öğrenme teknikleri:

Metinden ontoloji öğrenme (Ontology learning from texts)

Sözlükten ontoloji öğrenme (Ontology learning from dictionary)

Bilgi tabanından ontoloji öğrenme (Ontology learning from knowledge base)

Yarı yapısal şemadan ontoloji öğrenme (Ontology learning from semi-structured schema)

İlişkisel şemadan ontoloji öğrenme (Ontology learning from relational schemata)

Hemen hemen tüm teknikler iyi bir doğal dil işleme (NLP, natural language processing) bilgi gerektirmektedir. [51]

Ontoloji Kütüphaneleri ve Ortamları

- Birçok iyi tanımlanmış ontolojiye erişimimizin tanımlandığını düşünürsek, yeni bir ontoloji yaratmamızın var olan ontolojilerin birleşimine yeni özellikler ekleyerek olması kaçınılmazdır. Bu “tekrar kullanılabilirlik” işlevini kullanabilmemiz için iki ana araca ihtiyacımız vardır:

- Varolan ontolojilere erişme ve kaydetmeye yönelik araçlar

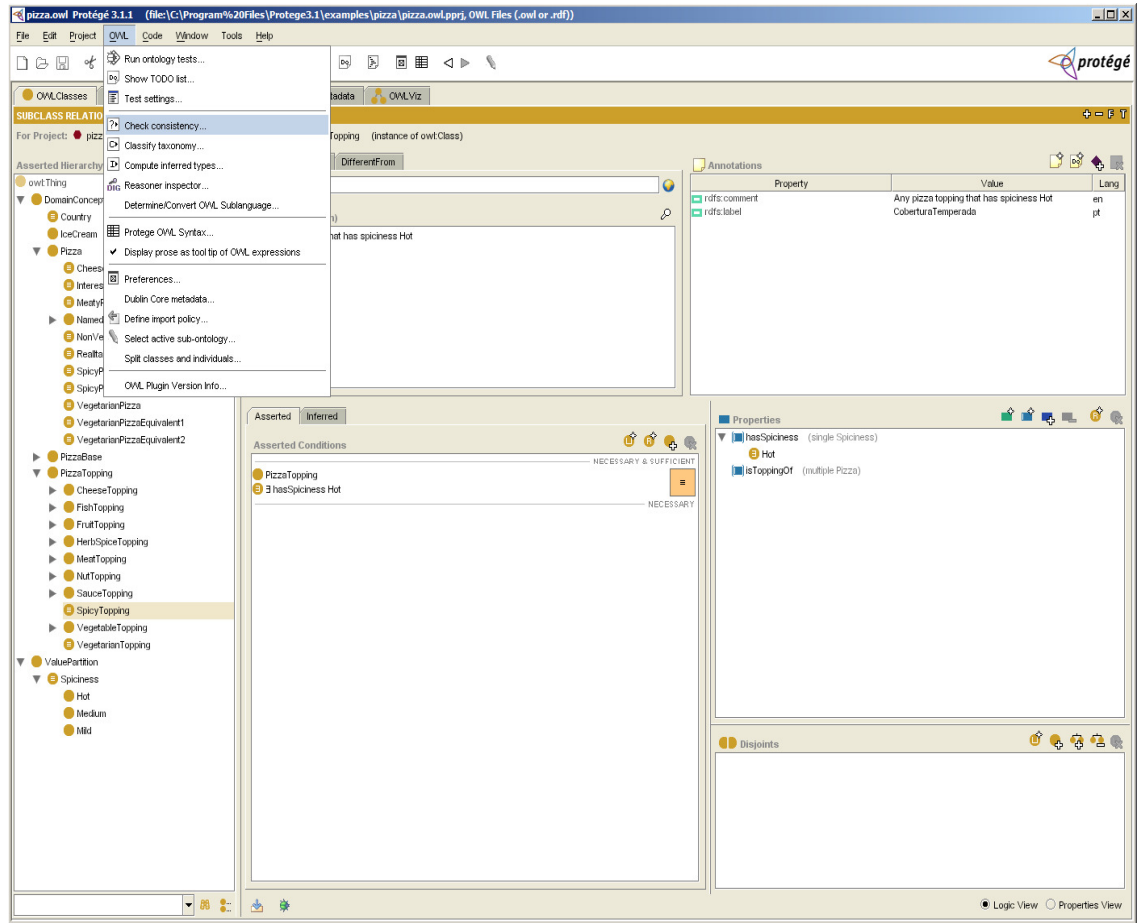
- Varolan araçları işleme ve yönetmeye yönelik araçlar

- Ontoloji kütüphane sistemlerine örnekler: DAML Ontology library system, SHOE, Ontology Server, IEEE Standard Upper Ontology, Sesame Veri Deposu gibi [52].

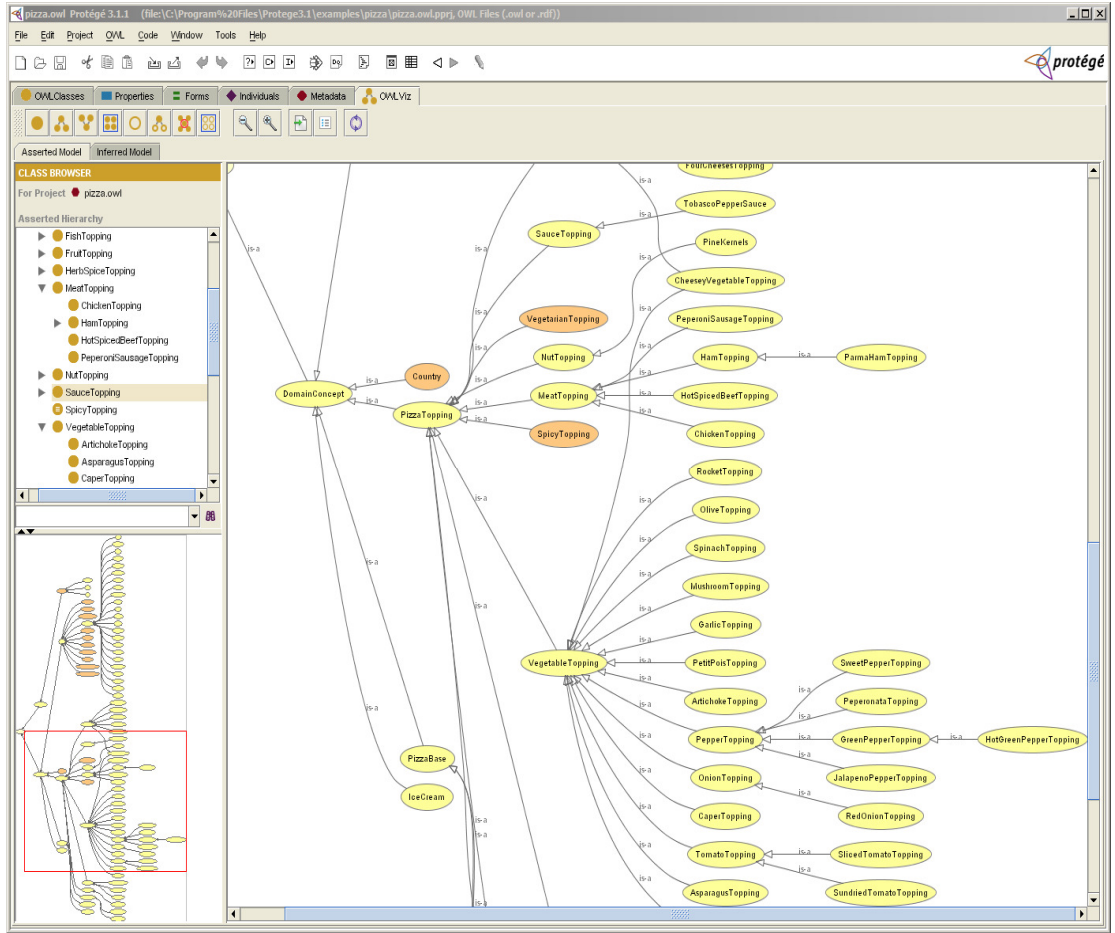
Ontoloji Editörleri

- Ontoloji editör araçlarının amacı, ontoloji dosyalarını tasarlamak, düzenlemek ve idame ettirmektir. Bazı örnekleri; Protégé, Ontolingua, WebOnto, WebODE, OntoEdit, OilEd'dir.

- Bunlardan Protégé, Stanford University School of Medicine tarafından geliştirilmiş, açık kaynak kodlu, platform bağımsız, farklı araçlarla entegre olabilen bir geliştirme ortamıdır. Oluşturulan ontolojinin anlamlandırma ve uyumluluk kontrolü için birtakım “çıkartım makineleriyle” birlikte çalışabilmektedir [53].



Şekil 4.1. Protégé ontoloji geliştirme aracı arayüzü



Şekil 4.2. Protégé ontoloji geliştirim aracı arayüzü-2

Ontoloji Yönetim Sistemi

- Bir ontoloji yönetim sistemi ile ontolojiler erişen ve onlar üzerinde sorgular yapan bir uygulamanın veri kaydetme ya da veri yapısının nasıl yapılacağı ile ilgili endişelenmesi gerekmez çünkü aynı “ilişkisel veritabanlarında” olduğu bu işleri ontoloji yönetim sistemi ele almaktadır.

- Ontoloji düzenleme özelliği, ontoloji yönetim sisteminin merkezi parçası değildir. Bazı sistemler, bir programlama arayüzü üzerinden ontolojileri programatik olarak düzenlemeyi sağlayan özelliklere sahiptir. Bu özelliklerin sağlanmadığı durumlarda ise geliştiriciler, Protégé gibi grafiksel düzenleme ortamlarını seçebilirler.

Jena Semantic Web Framework: Jena [54], programatik olarak Semantik Web uygulamaları geliştirmek için var olan bir Java framework’üdür. RDF, RDFS ve OWL ontolojileri için kural-tabanlı anlamlandırma(çıkarsama) motoru, programlama

ortamı oluşturur. Verilen bir ontoloji ve modeli için Jena'nın çıkarsama motoru anlamlandırma yapabilir böylece, modelin tanımlamadığı ek ifadeler türetilebilir. Ontoloji oluşturulduktan sonra, ontolojide tutulan bilgilerin bir ontoloji sorgulama dili ile sorgulanması sağlanmaktadır. RDF tabanlı herhangi bir ontolojinin Java kullanılarak modelinin oluşturulup üzerinde sorgulamanın ve çıkarsamanın yapılabilmesi için JENA çerçevesi kullanılabilir. RDF tabanlı bir ontoloji için RDF tabanlı bir sorgulama diline ihtiyaç vardır. Bunun için de JENA yazılım çerçevesi içerisinde yer alan RDQL sorgulama dili (RDF Query Language, <http://www.w3.org/Submission/RDQL/>, W3C) kullanılabilir.

4.6. Ontoloji Sorgulama Dilleri

Bilgisayar bilimleri ve yapay zeka olarak, ontoloji dilleri ontolojileri oluşturmak için kullanılan resmi dillerdir. Onlar belirli etki alanları hakkında bilgi kodlama sağlar. Genellikle bu bilgi işlem destek mantık kuralları içerir. Ontoloji dillerinde neredeyse her zaman çerçeve dili genellemeleri vardır, çoğunlukla bildirimsel dillerdir. Ya birinci derece mantığa ya da açıklama mantığına dayanmaktadır.

SPARQL:

SPARQL anlamsal web için bir sorgulama dili ve veri erişim protokolüdür. W3C tarafından RDF veri modeli için tanımlanmıştır. RDF sorgulama dilleri üzerindeki çalışmalar son birkaç yıldır devam etmektedir. Bu süreç içerisinde RDQL, Squish, Versa gibi farklı yaklaşımların kullanıldığı diller geliştirilmiştir. RDQL ve Squish'e benzer olarak SQL sözdizimini örnek alan bir dil olan SPARQL kendisine geniş bir kullanım alanı bulmuştur. RDF ve OWL sorgulama araçlarının büyük çoğunluğu SPARQL desteği sunmaktadır [55].

Örnek: SPARQL Sorgu Biçimi

ön ek bildirimleri

PREFIX foo: <http://example.com/resources/> ...

sonuç yancümlesi

SELECT ...

sorgu kalıbı

WHERE { ... }

sorgu değiştiricileri

Sıralama (Order By)

- URI'leri, yani biricik kaynak tanıtıcılarını kısaltmada kullanılan ön ek bildirimleri,
- Sorgudan elde edilmek istenen bilgilerin belirtildiği sonuç yancümlesi,
- Veri kümesi üzerinde yapılacak sorgunun belirtildiği sorgu kalıbı,
- Sıralama, bölme gibi sonuç kümesi üzerinde düzenleme yapmaya yarayan sorgu değiştiricileri.

Elementlere ait isim, sembol, renk, atom numarası ve kütle bilgilerinin yer aldığı periyodik tablo veri seti üzerinden, elementlere ait isim, atom numarası ve renk bilgilerini, atom numarasına göre sıralanmış biçimde çekebilmek için tanımlanmış örnek sorgu aşağıda verilmiştir.

PREFIX table:

```
<http://www.daml.org/2003/01/periodictable/PeriodicTable#
>
```

```
SELECT ?name ?number ?color
```

```
WHERE
```

```
{
```

```
    ?element table:name ?name.
```

```
    ?element table:symbol ?symbol.
```

```
    ?element table:atomicNumber ?number.
```

```
OPTIONAL { ?element table:color ?color. }
```

```
}ORDER BY ?number
```

SPARQL herhangi bir sorguyu, yapısal ya da yarı-yapısal RDF çizge-kalıpları üzerinden eşleştirme yolu ile sonuçlandırır. Periyodik tablo örneği için, bazı elementlerin renk bilgilerinin olmadığı bilinmektedir. Buna rağmen üstte belirtilen sorguda, renk bilgileri olmayan elementlerin de sonuç kümesi içinde yer alması OPTIONAL anahtar sözcüğü ile sağlanmaktadır. Benzer şekilde UNION, SPARQL

için önemli bir işleç olup, sonuç kümeleri üzerinde mantıksal VEYA işlemini gerçekleştirir.

SPARQL17 -SPARQL Protocol And RDF Query Language

- Üçlü deseni
- Özne, yüklem ve nesnesi değişken olabilen bir üçlü
- Değişkenler ? karakteri ile başlar
- Çizge deseni
- Birbirlerine zincirleme bağlanabilen birden fazla üçlü deseni

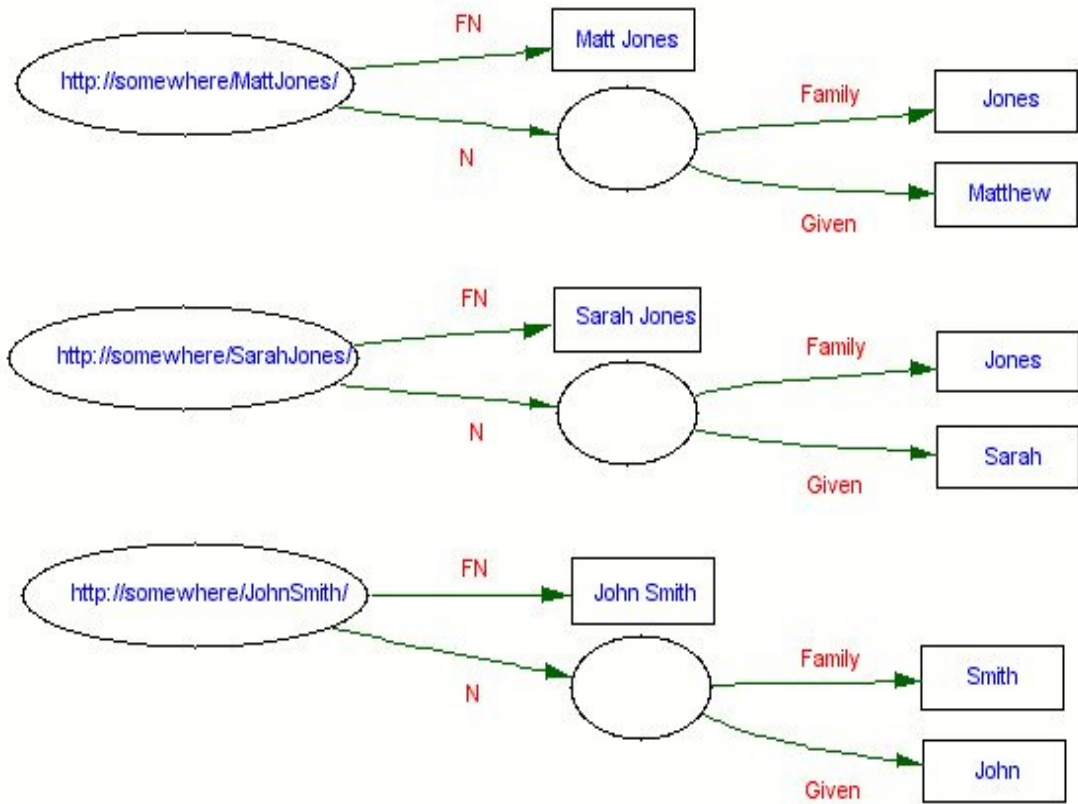
OWL-QL (OWL Query Language)

RDQL(RDF Data Query Language)

Basit Syntax;

```
SELECT değişkenler  
FROM dokümanlar  
WHERE ifadeler  
AND filtreler  
USING isimuzayı tanımları
```

Ontoloji Sorgulama Örneği



Şekil 4.3. Ontoloji Sorgulama Örneği

Sorgu 1:

```
SELECT ?x
```

```
WHERE (?x <http://www.w3.org/2001/vcard-rdf/3.0#FN> "John  
Smith")
```

Sorgular Jena çerçevesine sahip makinelerde komut satırından “java jena.rdfquery --data m1.rdf --query q1” girilerek çalıştırılabilir.

Çıktı 1::

```
<http://somewhere/JohnSmith/>
```

Sorgu 2:

```
SELECT ?givenName
```

```
WHERE (?y <http://www.w3.org/2001/vcard-rdf/3.0#Family>  
"Jones"),
```



```
(?y <http://www.w3.org/2001/vcard-
rdf/3.0#Given> ?givenName)
```

Çıktı 2:

givenName

=====

"Matthew"

"Sarah"

The screenshot shows the Sesame web interface. At the top, there is a navigation bar with 'File', 'Edit', 'View', 'Go', 'Bookmarks', 'Tools', and 'Help'. Below this is a status bar showing 'Logged in: - [log in]' and 'Repository: Museum demonstration [select other]'. The main content area is titled 'Evaluate a SeRQL-select query'. It contains a text input field for the query: 'Select * From (x)p(y) Where x LIKE "*rembrandt*" IGNORE CASE'. Below the input field, there are buttons for 'Clear', 'Append namespaces', and 'Evaluate'. The 'Response format' is set to 'HTML'. At the bottom of the interface, there is a 'Query results:' section displaying a table with three columns: 'x', 'p', and 'y'. The table contains 12 rows of results, each showing a URI for 'x', a property URI for 'p', and a resource URI for 'y'. The 'x' column contains URIs from 'http://www.european-history.com/rembrandt.html' and 'http://www.artchive.com/rembrandt/abraham.jpg'. The 'p' column contains URIs from 'http://www.w3.org/1999/02/22-rdf-syntax-ns#type', 'http://www.icom.com/schema.rdf#creates', and 'http://www.w3.org/1999/02/22-rdf-syntax-ns#vne'. The 'y' column contains URIs from 'http://www.w3.org/2000/01/rdf-schema#Resource', 'http://www.icom.com/schema.rdf#Artist', 'http://www.icom.com/schema.rdf#Painter', 'http://www.icom.com/schema.rdf#Flemish', 'http://www.artchive.com/rembrandt/abraham.jpg', 'http://www.artchive.com/rembrandt/artist_at_his_easel.jpg', 'http://www.artchive.com/rembrandt/abraham.jpg', 'http://www.artchive.com/rembrandt/artist_at_his_easel.jpg', 'http://www.w3.org/2000/01/rdf-schema#Resource', 'http://www.icom.com/schema.rdf#Artifact', and 'http://www.icom.com/schema.rdf#Painting'.

x	p	y
http://www.european-history.com/rembrandt.html	http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://www.w3.org/2000/01/rdf-schema#Resource
http://www.european-history.com/rembrandt.html	http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://www.icom.com/schema.rdf#Artist
http://www.european-history.com/rembrandt.html	http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://www.icom.com/schema.rdf#Painter
http://www.european-history.com/rembrandt.html	http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://www.icom.com/schema.rdf#Flemish
http://www.european-history.com/rembrandt.html	http://www.icom.com/schema.rdf#creates	http://www.artchive.com/rembrandt/abraham.jpg
http://www.european-history.com/rembrandt.html	http://www.icom.com/schema.rdf#creates	http://www.artchive.com/rembrandt/artist_at_his_easel.jpg
http://www.european-history.com/rembrandt.html	http://www.icom.com/schema.rdf#paints	http://www.artchive.com/rembrandt/abraham.jpg
http://www.european-history.com/rembrandt.html	http://www.icom.com/schema.rdf#paints	http://www.artchive.com/rembrandt/artist_at_his_easel.jpg
http://www.artchive.com/rembrandt/abraham.jpg	http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://www.w3.org/2000/01/rdf-schema#Resource
http://www.artchive.com/rembrandt/abraham.jpg	http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://www.icom.com/schema.rdf#Artifact
http://www.artchive.com/rembrandt/abraham.jpg	http://www.w3.org/1999/02/22-rdf-syntax-ns#vne	http://www.icom.com/schema.rdf#Painting

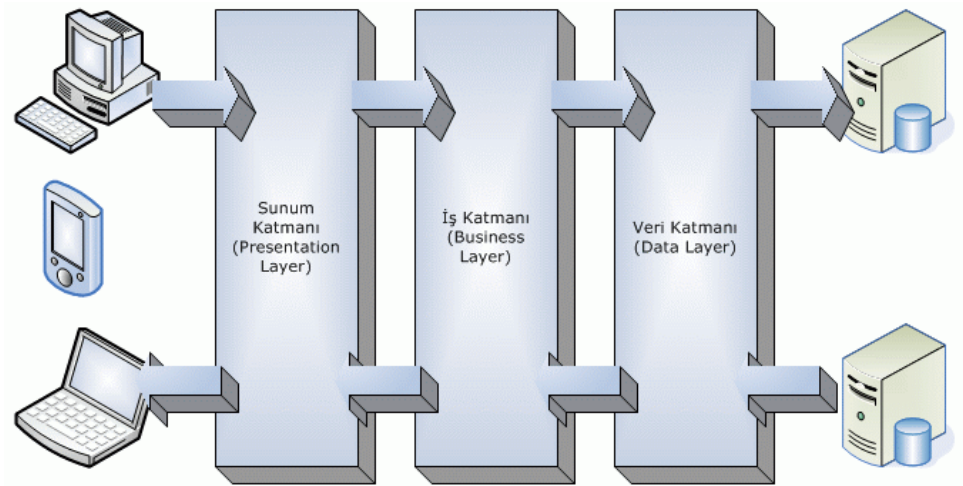
Şekil 4.4. Sesame veri deposu

5. UYGULAMA YÖNTEMİ

Sistem tasarımının ilk adımı, tasarım kararlarını gerçekleyen bir ontolojinin oluşturulmasıdır. İkinci adım oluşturulan ontoloji üzerinde istenen işlemleri yapabilecek bir altyapının oluşturulmasıdır. Son adım ise oluşturulan bu altyapıyı kullanarak ontoloji üzerinde gerekli işlemleri yapabilecek kullanıcı arayüzlerinin (Graphical User Interface) oluşturulmasıdır.

5.1. Semantik Web Uygulama Mimarisi

Semantik Web Uygulaması, Üç Katmanlı Veri-İş-Sunum mimarisini gerçeklemede olduğu için, bu konuda üç katmanlı mimari ve nesneye dayalı programlama desteğinden dolayı ASP.NET teknolojisi kullanılmaktadır. Üç Katman Mimarisi, özellikle nesneye dayalı programlama ile birlikte güç bulmuş olan ve veritabanı programlamanın esas dinamiklerini oluşturan bir programlama eğilimidir.



Şekil 5.1. Semantik web uygulaması katmanlı mimari [56]

Üç katman mimarisinin en alt katmanını, sistemde işlemekte olduğumuz veriyi tutan veritabanından oluşan veri katmanı oluşturur. Veri katmanı ile temasta bulunan, uygulamanın en alt katmanına iş katmanı denir ve prensip olarak nesne yönelimli programlamanın temelini oluşturan sınıflardan (class) oluşur. En üst katmanı oluşturan sunum katmanının görevi, iş katmanı ile kullanıcıyı etkileştirmektir.

Sunum katmanının doğrudan veritabanına erişimi bulunmamaktadır. Bu işlemler için, iş katmanında geliştirilmiş sınıfları kopyalandırarak kullanmaktadır. Uygulamamızın ayrı katmanlar halinde olmasının avantajı, yönetiminin ve güncellenmesinin parçala ve yönet alt yapısı kullanıldığından kolay ve hızlı olmasıdır.

Çizelge 5.1. Sistemin katmanlı mimarisi

Katman	İşlevi	Sistemde Yer Alan Bileşen Adları
Sunum	Kullanıcı ile etkileşim, iş katmanında kodlanmış nesnelere kopyalandırıp kullanmak.	Default.aspx.designer.cs KitapGoster.aspx.designer.cs
İş – Anlamsal Web Katmanı	Sunum ve Veritabanı katmanları ile etkileşim, iş kurallarının sağlanması.	Default.aspx.cs KitapGoster.aspx.cs Kitap.xml ontoloji dosyası.
Veri	Tablolar, Stored Procedure'ler, Kısıtlar ve diğer veritabanı bileşenleri ile kurgulanmış veri mantığının sağlanması.	SWP.dbo veritabanı altındaki "Kitap" tablosu. Veritabanının bir özetini oluşturan LINQ altyapısındaki "Project.dbml" dosyası.

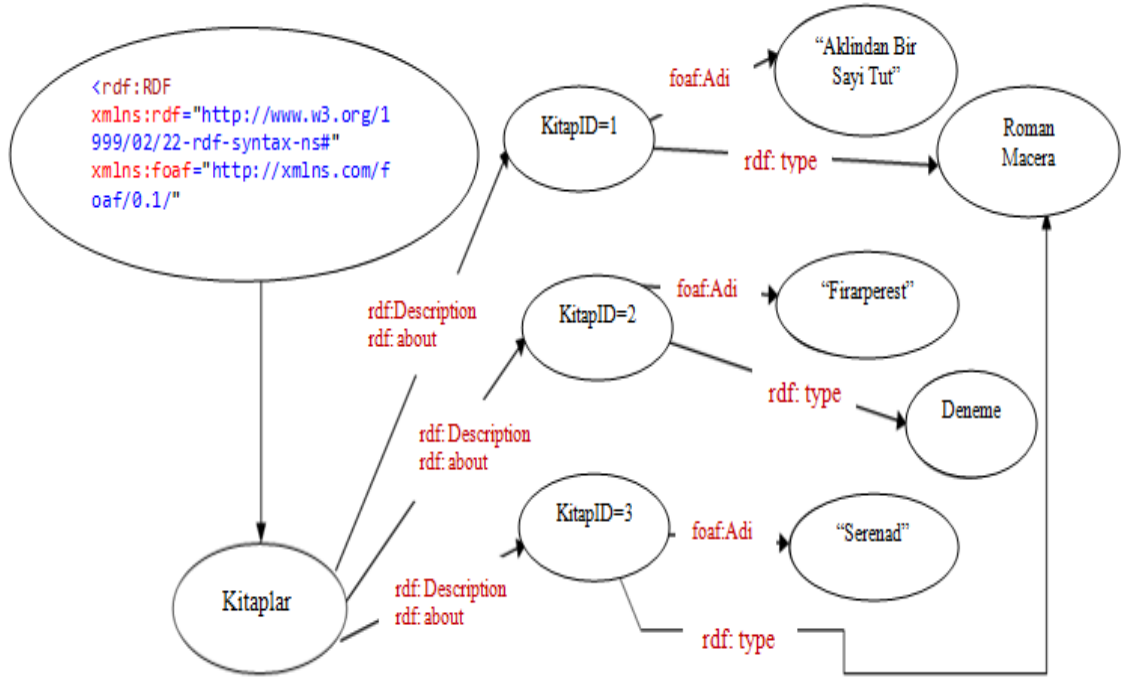
Sistemimizin geliştirme süreci üç ana adımdan oluşmaktadır:

- Ontolojinin geliştirilmesi
- Oluşturulan ontolojinin yönetilmesi ve sorgulanmasına yönelik programlamanın geliştirilmesi
- Kullanıcı arayüzlerinin tasarlanması

5.2. Ontoloji Oluřturma

Ontolojiler, Web Ontology Language (OWL) [57] dili kullanılarak geliřtirilir. Belirli sözdizim özelliklerine sahip olan OWL dilinin temelini XML yapısı oluşturur. Tezimizde tasarım kararı olarak ontoloji dosyamız belirlenen söz dizime uyan XML yapısında elle oluşturularak geliřtirilmiřtir. Ontoloji elemanları ve bunlar arasındaki iliřkilerin takibi yapılmaktadır. Bu řekilde, “kitap.xml” dosyasına eklenen her yeni kitap kaydında XML dosyası güncellenmektedir ve dosya alt yapısı için belirlenmiř taslak halinde yeni eklenen kitap özellikleri (biricik numara, ad, tür) tanımlanmaktadır. Bu tanımlar sistemin SWP (Semantic Web Project) veritabı ile aynı olacak řekilde oluşturulmaktadır. Geliřtirim ařamasında veritabanı ile ontoloji dosyasında uyumsuzluk oluřursa ilgili kayıt bilgilerinin getirimi saęlanmayacaktır.

Ontolojinin oluřturulmasından sonraki adım; bu ontolojiyi iřleyecek (sorgulama, kayıt, vb...) altyapının oluřturulmasıdır. “Kitap.xml” ontoloji kodu incelendięinde, elimizdeki metaverinin URI (Uniform Resource Identifier – Nizami Kaynak Belirteci)’lerle tanımlanan bir takım kaynaklar topluluęu olduęu görölmektedir. Bunun anlamı, bir veri yapısını (örneęin bir kitap nesnesini) kodsız olarak okuduęumuzda, elimize o veri yapısının tamamı (içerdięi dięer özelliklerle birlikte) yerine sadece ilgili URI ile tanımlanan bir kaynak (OWL kaynaęı) gelecektir. Bu durumda, bizim için anlamlı olacak bir etkinlik veri yapısını oluřturabilmek için, ontoloji modelindeki kaynaklar arasında URI’leri kullanarak birebir gezinilecektir.



Şekil 5.2. “Kitap.xml” dosyası altyapısı

World Wide Web normalde insan tüketimi için oluşturulmuştur ve içindeki her şey makine-okunurluğu olup makine-anlamlandırılabilirliği olmayan nesnelere. Web üzerinde herhangi bir şeyi otomatikleştirmek çok zordur çünkü Web’in içerdiği verinin hacmi çok büyüktür, bu nedenle de bu veriyi elle yönetmenin imkanı yoktur. Bu noktada RDF’in önerdiği çözüm Web’de bulunan veriyi tanımlamak için metadata kullanmaktır. Metadata “veri hakkında veri” anlamına gelmektedir (Örneğin, bir kütüphane katalogu bir metadattır çünkü kütüphanenin içerdiği kitaplar hakkında bilgi içerir.) ya da “Web kaynaklarını tanımlayan veri” tanımı da kullanılabilir. Tezimizde oluşturulan “kitap.xml” dosyası RDF diliyle/etiketleriyle yazılarak tutulmuş “kitap IDleri”, “kitap isimleri”, “kitap türleri” ile ihtiyacımız olan gerekli anlamlandırılmış veriyi metadata özelliğinde içermektedir. Normalde veritabanında kayıt ettiğimiz kitaplar hakkında tuttuğumuz verinin “adı”, “yazarı”, “türü”, “açıklaması”, “fiyatı” ve “yılı” bilgileri tutulmaktadır ancak RDF dosya yapısıyla kitapların veritabanında tutuldukları yerlerin bağlantıları tanımlanarak sorgu sonucu olarak RDF üzerinden kitap bilgilerine erişim sağlanmaktadır. Veri ve metadata arasındaki fark kesin değildir, bu fark spesifik bir uygulama tarafından

gerektiğinde sağlanmaktadır ve çoğu zaman da aynı kaynak her iki yolla da eş zamanlı yorumlanmaktadır.

RDF, metadatayı işlemeye vakıftır; Web’de makine-anlamlandırılabilirliği olan bilgiyi deęiş tokuş yapan uygulamalar arasındaki birlikte işlerliği sağlamaktadır. RDF Web kaynaklarının otomatize işlenmesine olanak vermenin önemini vurgular. RDF bir çok uygulama alanında kullanılabilir; örneğin tezimizde olduđu gibi kaynak keşfinde daha iyi arama motoru yeteneđi sunmada, bir Web sitesindeki, sayfadaki ya da dijital kütüphanedeki içeriđi ve içerikler arası ilişkileri tanımlamak için akıllı yazılım ajanlarının bilgi paylaşımı ve deęişimini kolaylaştırmasının yardımıyla kataloglamada, içerik puanlamada, tek bir matiksal dokümanı temsil eden sayfalar koleksiyonunu tanımlamada, bir Web sitenin güvenlik politikasını olduđu kadar bir kullanıcının da güvenlik tercihlerini tanımlamada kullanılmaktadır [58].

5.3. Oluşturulan Ontoloji Üzerinde İşlemler

Sistemde oluşturulmuş ontoloji dosyası üzerinde yapılan sorgulama işlemleri için kullanılacak isim uzayları ve kütüphaneler aşağıdaki bölümlerde anlatılmaktadır.

5.3.1. SemWeb kütüphanesi

DOT NET’in SemWeb kütüphanesi, Mono veya Microsoft’un .NET 1.1/2.0 için olan C# dilinde yazılmış açık-kaynaklı bir kütüphanedir. Kütüphane RDF (XML, N3) okumak ve yazmak için, RDF’i kalıcı depolamada (bellek, MySQL, vs) tutmak için, kalıcı depolamayı basit grafik eşleştirmesiyle ve SPARQL ile sorgulamak için ve SPARQL sorgularını uzak son-noktalardan gerçekleştirmek için kullanılır. Sınırlı RDFS ve genel-amaçlı anlamlandırma da mümkündür. SemWeb’in API’si güvenilir ve esnektir.

Kütüphanedeki temel sınıflar SemWeb namespace’inde bulunmaktadır. Bu isimuzayında dört sınıf kütüphanenin tüm temel işlevlerini sağlamaktadır: Resource, Statement, StatementSource ve StatementSink.

SemWeb ilk defa Haziran 2005'te yayınlanmıştır ve 1 milyarın üzerinde ifadeyi içeren paketlerle üçerli test edilmiştir. Ana özellikleri, RDF'I XML ve N3'te okumak/yazmak, kalıcı SQL-destekli depolama ve SPARQL sorguları gibi, çok sağlamdır. Çevresel özellikleri, RDFS anlamlandırma ve geri-zincirleme anlamlandırma, çalışmaktadır fakat en az test edilmiş ve en az tamamlanmış olan kısımlardır. Kütüphanenin OWL şema için özel bir aracı yoktur. Sadece RDF üçlüleri seviyesinde işlem yapar.

SPARQL desteği, (bir kaç hata düzeltimi ve performans iyileştirmesi ile birlikte) Ryan Levering'in Java'da SPARQL uygulamasını IKVM (IKVM Jeroen Frijters tarafından geliştirilmiştir) ile .NET'e çevrimine dayanır. Genel-amaçlı anlamlandırma için olan Euler sınıfı Jos De Roo'nun Euler anlamlandırma makinesi için uyarladığı JavaScript'idir.

SemWeb, GNU GPL (versiyon 2 veya üzeri) altında lisanslanmıştır fakat parçaları GNU GPL ve Creative Commons Attribution olmak üzere çift lisanslıdır [59].

Özellikler:

- Kolay API: uygulaması kolay ve tamamen “çarpraz platform” özelliğinde.
- RDF/XML: RDF/XML Okuma ve Yazma (XMP dahil). Okuyucu gruplara ayırır, bütün dökümanın yüklenmesi hiçbir zaman gerekli olmaz. Sözdizimsel analiz W3C testlerini işler.
- Notation 3: Okuma ve yazma NTriples, Turtle, birçok Notation 3 (bunların hepsi kabaca verileri gruplara ayırır) 20,000 ifade/saniye).
- IRIs ve XSD doğrulaması; veri türleri aslına uygundur dosyaların okunması boyunca (veya okunması istenildiği zaman) ve xsd:date Time sözdizimsel analiz, tarih, ve saat XsdDateTime aşamasına girer.
- Çıktılar GraphViz dot format içindedir.
- SQL DB-destekli kalıcı depolama SQL Server için, MySQL, Sqlite, and PostgreSQL. MySQL en azbir milyon üçlüyü ölçeklendirebilmektedir.
- Bellek-içi depolamaya olanak vermektedir.
- Kalıcı Bellekte Gelişmiş Seçim operasyonunu destekler; birçok veriyi bir kere de sorgulaması için (individual calls yapmaktan çok daha hızlı bir şekilde).

- Çıkarsama: RDFS sonuca varır (tam anlamıyla olmamasına rağmen) ve kural tabanlı anlamlandırma ve ters-zincirleme Euler makinesine dayanmaktadır.
- 4-Tuples: Aşamalar üçlü değil, dördü şeklinde yapılmaktadır. Dördüncü meta alanı uygulamaya yönelik amaçlar için kullanılır. Örneğin; kaynak depolamak, ifadeleri gruplamak veya N3 formülasyonları depolamak gibi.
- Sorgulama: SQL mümkün içine sorguları çevirileri ile Basit grafik gerektirme testleri ve herhangi bir veri kaynağı üzerinde SPARQL sorgular. Uzak SPARQL veri kaynağı istemci de mevcuttur. Ve bir ASP.NET SPARQL Protokolü sunucu işleyici vardır.
- Sadece SelectableSource nesnelere addsource ile eklenmiş durumdadır bir Store örneğinde herhangi bir sorgu çalıştırarak federe veri kaynakları üzerinde sorgular.
- Genişletilebilirlik: yeni kalıcı depolamaları veya kaynakları ifadelerin uygulanması gibi bir arayüz, ya salt ileri ya da sürekli olarak uygulanması kadar basittir.
- Deneysel mesaj bulma ve grafikleri yalın yapmak için algoritmalar kullanılır [60]

```
Statement s in store.Select(new Statement(null, rdftype,
KitapTur))
```

Burada, SemWeb kütüphanesinin “public Statement (Entity subject, Entity predicate, Resource object)” methodu kullanılmaktadır.

```
Resource r in store.SelectObjects(s.Subject, foafAdi)
```

Burada, SemWeb kütüphanesinin “protected virtual void SelectObjects (SemWeb.Entity subject, SemWeb.Entity predicate)” methodu kullanılmaktadır.

5.3.2. Friend of a friend (FOAF)

```
Entity foafAdi = "http://xmlns.com/foaf/0.1/" + "Adi";
```

```
Entity rdftype = "http://www.w3.org/1999/02/22-rdf-syntax-ns#" + "type";
```

Burada kullanılan Entity nesnesini daha iyi açıklamak için SemWeb isim uzayının Kaynaklar (Resources) ve İfadeler (Statements) kavramlarından bahsedilecektir. Kaynak, RDF’teki terimlerin soyut bir alt sınıfıdır. RDF resmi modelinin iki tipte

terimi vardır; düğümler ve sözsözsel değerler. Ve benzer şekilde bu alt sınıfın iki alt sınıfı bulunmaktadır: Entity ve Literal. Düğümler, isimlendirilmiş olsunlar (Örneğin URI'ler) ya da olmasınlar (isimsiz, boş), Entity sınıfı yardımıyla gösterilirler. İsimli düğümler Entity nesneleriyle direk olarak gösterilirken, boş düğümler ise BNode sınıfı yardımıyla gösterilirler. BNode sınıfı Entity sınıfının bir alt sınıfıdır. Sözsözsel değerlerin gösterimi ise Literal sınıfı ile yapılmaktadır.

SemantikWeb2011 yazılımımızda kullandığımız “kitap.xml” dosyamız, FOAF altyapısı kullanılarak ve FOAF yapısının kitap isimlerini baz alması suretiyle kullanılmaktadır. Kurmakta olduğu ilişkileri “kitap.xml” dosyasında RDF etiketleri içerisinde almaktadır.

FOAF, Web'in kendisi gibi, bağlantılı bir bilgi sistemidir. Semantik Web teknolojisi kullanılarak oluşturulmuştur ve çeşitli uygulamalar, Web siteleri, servisleri ve yazılım sistemleri üzerindeki verinin birleşimine izin verecek şekilde tasarlanmıştır. Bunu başarmak için FOAF, veri deęiş tokuşuna kolaylık saęlayan bir yaklaşım izlemektedir. Kullanıcının kendisi ya da başkaları hakkında önceden tanımlama yapmasına gerek yoktur. Bu spesifikasyon, insanlar ve onların yaptıkları hakkında konuşmak üzere kullanılan temel bir “sözlük” terimlerini içermektedir.

FOAF, “şemalar” veya “ontolojiler” gibi dięer sözlükleri de kullanmak üzere ve Semantik Web için yaratılmış çok çeşitli genel araçların ve servislerin tekrardan kullanılabilmesi için tasarlamıştır. Örneğin, SPARQL üzerinde çalışan W3C bize FOAF verisinin veritabanına danışabilmek için zengin bir sorgudili saęlamaktadır. Dięer W3C grupları da, her çeşitte RDF verisini Web sayfaları içinde kodlamak için gelişmiş mekanizmalar üzerinde çalışmaktadırlar. Semantik Web bize işbirliği için mimari sunmaktadır ve zayıf-koordinasyonlu geliştirmeciler tarafından kompleks teknik kavramların paylaşılabilmesini saęlamaktadır.

FOAF projesi, makine-okunurluęu olan Web sayfaları üzerinde, insanlar, gruplar, şirketler için tanımlama yapabilme mantığı temellidir. Bunu başarmak için “FOAF kelimeleri” kullanılmaktadır. FOAF projesinin kalbinde, dünya hakkındaki

açıklamaları ifade edebilmek amacıyla terimlerin bir sözlüğü şeklinde davranması için tasarlanmış tanımlama kümesi yatmaktadır.

Entity KitapTur = Kullanıcının arayüz üzerinden seçtiği Kitap Türünü <http://localhost:52497/kitap.xml> lokasyonlu XML dosyasının içinde tutacağı belirtmektedir.

5.3.3. MemoryStore sınıfı

MemoryStore sınıfı, ifadeler için bellekte depolamayı sağlamaktadır. Özünde, ifadelerin sadece bir DiziListesidir. Bu sınıf (ve bütün Store sınıfları) şimdiye kadar tartışılan sınıf hiyerarşisi bakış açısından farklıdır: hem StatementSource hem de StatementSink uygulamaktadır. Bu sayede, add methodunu çağırarak ifadeler eklenebilmektedir. Aynı zamanda select (StatementSink) methodunu çağırarak ifadeler herhangi bir StatementSink içine getirilebilmektedir [61]. (Örneğin, bu bir MemoryStore, dosya-yazım sınıfı veya kullanıcının yazdığı sınıflardan biri olabilir.)

İfadelerin alındığı yer StatementSource adını almaktadır. Bazı ifadeleri belli bir nesneye yüklemeyi amaçlayan ve select methoduna sahip olan bir arayüzdür. Yükleme yapılan nesne StatementSink adını almaktadır. İlgili ifadelerin eklenmesi için add method kullanılmaktadır.

```
Store store = new MemoryStore();
```

Burada, kitap bilgilerini tutacak nesne olarak MemoryStore sınıfından türetilmiş bir değişken olan store kullanılmıştır.

5.3.4. XmlDocument sınıfı

XmlDocument sınıfı, W3C Document Object Model (DOM) Ana Seviye 1 ve Seviye 2'yi uygulamaktadır. DOM, XML dokümanının bellekteki ağaç gösterimini oluşturur ve bu dokümanda gezinmeye ve güncellemeye izin verir. XmlDocument sınıfı

IXPathNavigable arayüzünü uygulamakta olduğu için, sınıf aynı zamanda XsltTransform sınıfı için kaynak doküman olarak da kullanılmaktadır.

XmlDataDocument sınıfı XmlDocument sınıfından türemektedir ve yapısal bir verinin ilişkisel bir DataSet'i içinde kaydedilmesi, geri getirilmesi ve işlenmesini sağlamaktadır [62].

```
System.Xml.XmlDocument doc = new
System.Xml.XmlDocument();
```

Burada “doc” değişken isminde XML dokümanı yaratılmaktadır.

```
doc.Load("http://localhost:52497/kitap.xml");
```

Kitap bilgilerinin hangi bölgede olduğu gösterilmektedir. XmlDocument sınıfının sahip olduğu Load(XmlReader) methodu yardımıyla, XML dokümanı belirlenen XmlReader'dan yüklenmiştir [62].

Sistemimizde arama yapılırken girilen sorgu kriterine göre önce RDF dosyasındaki kayıtlar taranmaktadır. Böylece performans açısından her bir sorgu isteğinde veritabanına gidilme süresi ortadan kalkmaktadır. Arama süreci ilk olarak RDF ontoloji dosyamızdaki (kitap.xml) satırlar üzerinden kitap ID'leri yardımıyla kısa sürede yapılmaktadır, daha sonra da bulunan kitaplar arasında ayrıntıları görüntülenmek istenen kitap var ise bu işlemde veritabanına gidilmekte ve gerekli kitap bilgileri çekilmektedir. Bu işlemde de Bölüm 5.3.6'da anlatılan LINQ altyapısı kullanıldığı için her bir sorguda veritabanı bağlantısı (Connection) açılıp kapatılmamaktadır, LINQ yapısının sistemimizin veritabanının bir sınıfını oluşturuyor olmasından dolayı zaman performansı açısından avantajlı duruma geçmekteyiz.

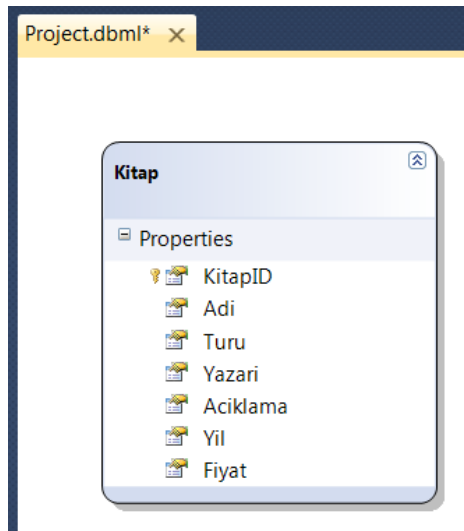
5.3.5. RdfXmlReader sınıfı

```
store.Import(new RdfXmlReader(doc));
```

Burada, kitap özet bilgileri nesneye yüklenmektedir ve RdfXmlReader sınıfını kullanarak, SemWeb isimuzayını kullanan bu sınıfın, RDF/XML dosyasından RDF ifadelerini okuma özelliğinden yararlanılmıştır. [63]

5.3.6. System.Linq isim uzayı

LINQ (Language Integrated Query), dil-entegrasyonlu sorgu, atama ve dönüştürme işlemlerini kuşatan .NET çerçevesinin bir uzantısıdır. C# ve Visual Basic dillerini ana dil sözdizimi ile sorgular için uzatır ve bu özelliklerden sınıf kütüphanelerinin yararlanmasına olanak sağlar. LINQ, C# 3.0 beraberinde kullanıcıların hizmetine sunulmuştur. Bu yapı ile C# dilinde koleksiyon bazlı yapılar üzerinde SQL sorguları tarzında ifadeler çalıştırılabilir hale gelmiştir. Bu sorgular, IEnumerable<T> türünden nesnelere ve XML yapısındaki veriler üzerinde işlem yapabilmemizi sağlamaktadır [64].



Şekil 5.3. “Project.dbml” dosyası

System.Linq isim uzayının kullanılmasının nedeni, bu sınıfın Language-Integrated Query (LINQ) (Dil-Birleşimli Sorgu) kullanan sorgulara destek veren bir arayüzü olmasıdır. System.Linq isim uzayı System.Core çeviricisi içinde yer almaktadır.

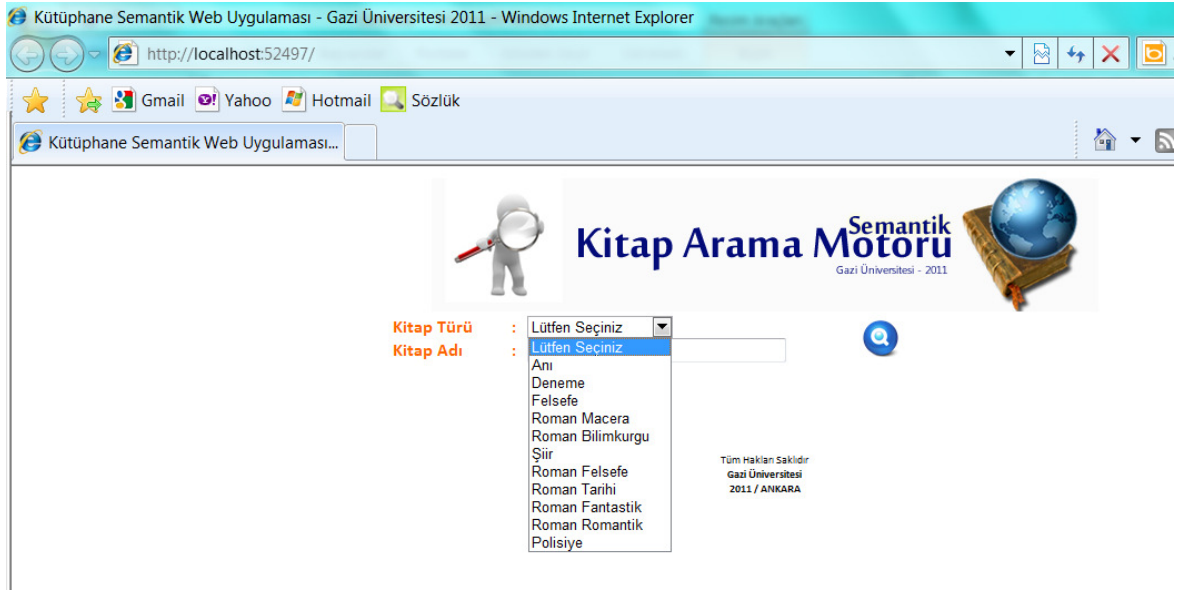
5.4. SemanticWeb2011 Yazılımı Arayüzleri

Kütüphane Semantik Web Uygulaması sistemimizin “Giriş Sayfası” kullanıcı arayüzü Şekil 5.4.’de gösterildiği gibidir.



Şekil 5.4. Giriş ekranı

Şekil 5.5.’de gösterilmekte olan arayüz üzerinden kullanıcılar Anı, Deneme, Felsefe, Roman Macera, Roman Bilimkurgu, Şiir, Roman Felsefe, Roman Tarihi, Roman Fantastik, Roman Romantik ve Polisiye türleri arasında sorgulama yapacakları türü seçebilmektedirler. Sayfa navigasyonu özelliği kapsamında herhangi bir sayfadayken logo kısmına basıldığında Giriş Sayfası görüntülenmektedir.



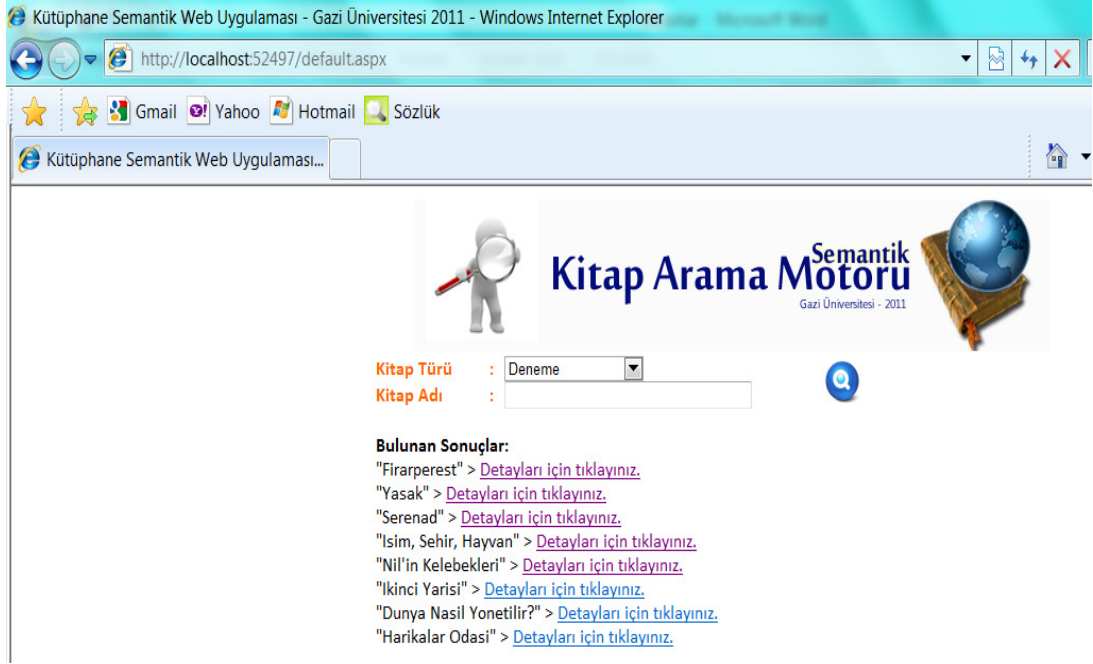
Şekil 5.5. Kitap türleri

Her iki kriterden hiçbiri doldurulmadan Arama butonuna basıldığında Şekil 5.6.'daki uyarı metni görüntülenmektedir.




Şekil 5.6. Kriter girişi

Şekil 5.7.'de gösterilen ekranda “Deneme” kitap türündeki kitaplar sorgulanmaktadır. Kriterlerden en az birinin doldurulması sorgu sonucunun dönmesi için yeterli olmaktadır.



Şekil 5.7. Sorgu sonucu örneği

Şekil 5.8.'de "Fırarperest" adlı kitabın detaylarını görüntüleyen "Kitap Göster" ekranı karşımıza çıkmaktadır. Sayfanın en sonundaki "Home" butonu ile Giriş Sayfasına dönüş yapılabilmektedir.



Kitap Arama Motoru
Semantik
Gazi Üniversitesi - 2011


Kitap Hakkında

Adı : Fırapperest
Yazarı : Elif Şafak
Türü : Deneme
Yılı : 2010

Tadına doyumaz, kimi zaman kışkırtıcı, kimi zaman sakinleştirici ama ruhu hep özgür kalan yazılar... İnsan ki eşrefi mahluktur, içindeki semavi özü keşfetmekle yükümlüdür. Çıkacaksın yollara, kendine doğru git gidebildiğin kadar. Keşif boynumuzun borcudur. Kendimizi keşfetmek, aşkı keşfetmek, dünyayı keşfetmek, Öteki"ni keşfetmek... (...) Çakılı kalmamak sırf alışkanlıklardan ötürü demir attığın koylara. Çıkmak oralardan, geçmek dalgakıranların beri tarafına, bilmediğin memleketlere varmak, tatmadığın yemekler yemek, sözlerini anlamadığın şarkılarla içlenmek, risk almak, dağılmak ve parçalanmak ve hasret çekmek buram buram, gurbetin tadına bakmak ve kendini yabancıların gözünden görmek, şaşırarak yeniden, şaşırarak bir çocuk gibi dünyanın hallerine, çeşitliliğine, güzelliğine, acımasızlıklarına... şaşırarak ölene kadar... şaşırma kabiliyetini hiç yitirmemek... budur son tahlilde Âdemoğullarına, Havvâkızlarına kendilerini keşfettiren serüven. Yazar Hakkında Strasbourg doğumlu Elif Şafak, çocukluğunu ve gençliğini Ankara, Madrid, Amman, Köln, İstanbul, Boston, Michigan

Açıklama : ve Arizona'da geçirdi. ODTÜ Uluslararası İlişkiler Bölümü'nü bitirdi, yüksek lisansını aynı üniversitede Kadın Çalışmaları Bölümü'nde, doktorasını ise siyasetbilimi alanında tamamladı. İlk romanı Pinhan'la 1998 Mevlâna Büyük Ödülü'nü aldı. Bunu Şehrin Aynaları (1999) ve Türkiye Yazarlar Birliği Ödülü'nü kazandı. Mahrem izledi (2000). Ardından her ikisi de çok satan ve geniş bir okur kitesine ulaşan Bit Palas (2002) ve İngilizce kaleme aldığı Araf (2004) yayımlandı. Med-Cezir'de (2005) kadınlık, kimlik, kültürel bölünme, dil ve edebiyat konulu yazılarını topladı. 2006'da senenin en çok okunan kitabı olan Baba ve Piç yayımlandı. Ardından aylarca satış listelerinden inmeyen ilk otobiyografik kitabı Siyah Süt'ü yazdı. Doğan Kitapçılık tarafından 2009 martında yayımlanan Aşk Türk yayıncılık dünyasında önemli bir rekora imza atarak, en kısa sürede en çok satan roman oldu. Tüm eserlerinden seçkiler niteliğinde olan Kâğıt Helva aralık 2009'da yine Doğan Kitapçılık tarafından yayımlandı. Eserleri otuz dile çevrilen Elif Şafak'ın romanları dünyanın en önemli yayınevlerinden Farrar, Straus and Giroux, Viking ve Penguin tarafından yayımlanmaktadır.

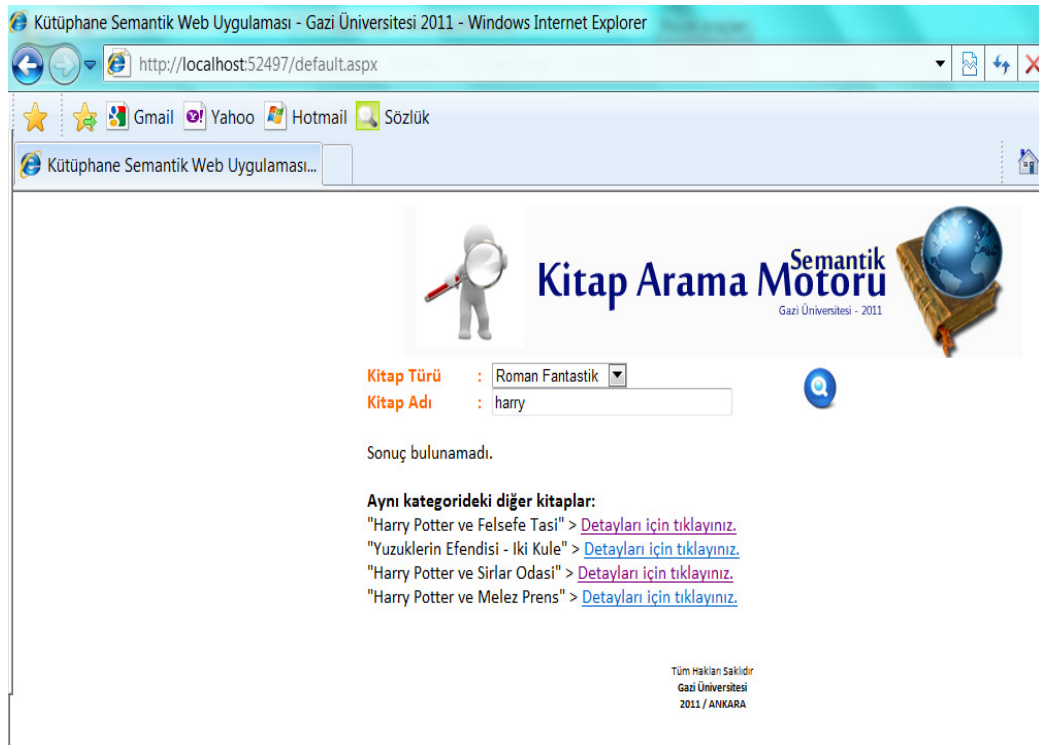
Fiyatı : 11,20 TL



Tüm Hakları Saklıdır
Gazi Üniversitesi
2011 / ANKARA

Şekil 5.8. Kitap ayrıntıları ekranı

Şekil 5.9.'da herhangi bir türde bir anahtar kelime ile (Kitap İsmi) arama yapıldığında ve ilgili kitap isminin olmadığı durumda o kategorideki diğer kitapların kullanıcıya ilgili olabileceği amacıyla listelenmesi senaryosu verilmektedir. Semantik Web altyapısının getirileriyle bu sorguda sonuç dönmeme gibi olumsuz bir durumla karşılaşmamaktadır. Kitap İsmi kısmına girilen kelime sonrasında sağda görülen "Arama Butonu" ile arama yapılabilirken, klavyeden "Enter" tuşuna basılması ile de arama yapılabilir.



Şekil 5.9. Sorgu sonucu olumlu değil

Kitap Adı kriterinin girilmesiyle yapılan arama işleminde girilen kitap adı tüm kitap türleri arasında aranmaktadır ve dönen sonuçlar Şekil 5.10.'da gösterildiği gibi oluşmaktadır.



Şekil 5.10. Sorgu sonucu olumlu

Bu tezde Kitap Adı kriterinin yanlış ya da eksik girilmesi ile yapılan arama işleminin sonuçlarında anlamsal olarak girilen anahtar kelimelere en yakın kitap listesi farklı türlerde olsa dahi getirilmekte ve Şekil 5.11., Şekil 5.12., Şekil 5.13.'de gösterilmekte olduğu gibi ekranlar kullanıcıya sunulmaktadır.



Kitap Türü : Lütfen Seçiniz

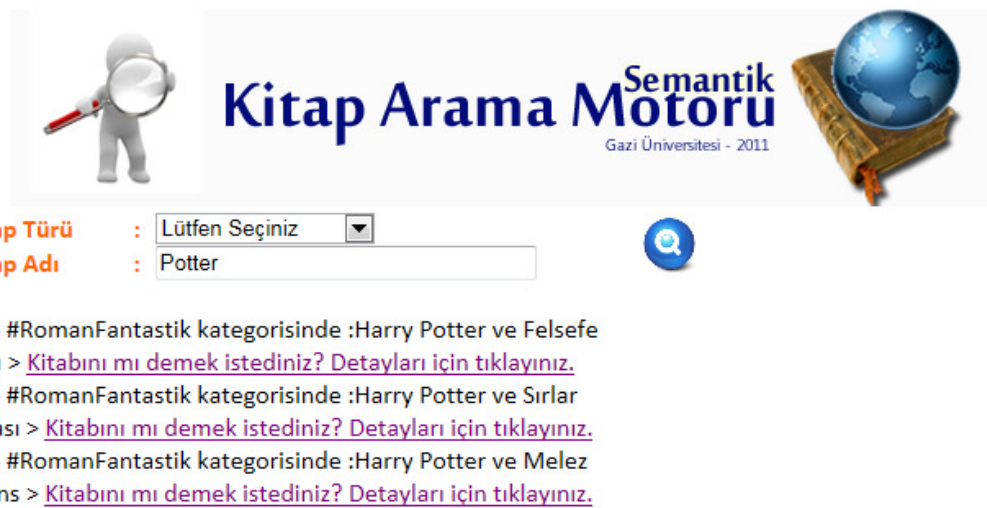
Kitap Adı : Serend

👉 - #RomanMacera kategorisinde :Serenad > [Kitabını mı demek istediniz? Detayları için tıklayınız.](#)

👉 - #Deneme kategorisinde :Serenad > [Kitabını mı demek istediniz? Detayları için tıklayınız.](#)

Tüm Hakları Saklıdır
Gazi Üniversitesi
2011 / ANKARA

Şekil 5.11. Semantik kelime tamamlama örneği - 1



Kitap Türü : Lütfen Seçiniz

Kitap Adı : Potter

👉 - #RomanFantastik kategorisinde :Harry Potter ve Felsefe Taşı > [Kitabını mı demek istediniz? Detayları için tıklayınız.](#)

👉 - #RomanFantastik kategorisinde :Harry Potter ve Sırlar Odası > [Kitabını mı demek istediniz? Detayları için tıklayınız.](#)

👉 - #RomanFantastik kategorisinde :Harry Potter ve Melez Prens > [Kitabını mı demek istediniz? Detayları için tıklayınız.](#)

Şekil 5.12. Semantik kelime tamamlama örneği - 2



Kitap Türü : Lütfen Seçiniz

Kitap Adı : Nil'in Kelebkleri

[#Deneme kategorisinde :Nil'in Kelebkleri > Kitabını mı demek istediniz? Detayları için tıklayınız.](#)

Tüm Hakları Saklıdır
Gazi Üniversitesi
2011 / ANKARA

Şekil 5.13. Semantik kelime tamamlama örneği - 3

6. SONUÇLAR

Anlamsal web, anlamsal web dilleri ve araçları şu anda çok aktif bir araştırma alanıdır. Bu konuda yapılan araştırma çalışmaları, anlamsal web (ontoloji) dilleri, ontoloji editörleri, araçları ve sistemleri, ontoloji sistem kütüphaneleri, değişik alanlar için geliştirilmekte olan ontoloji çalışmalarını kapsamaktadır.

Anlamsal Web ve Ontoloji araçlarının gelişmesi, bu alandaki uygulamaların artması, iş süreçlerinde, özellikle de B2B e-ticaret alanında yeni eğilimlerin ortaya çıkmasına yardımcı olacaktır. Web ortamında, bilgiler arasındaki ilişkilerin sağlanabilmesi ve bu bilgilerden anlamsal ilişkilerin çıkarılması, ekonomik çıktıyı yükseltecek ve önemli bir katma değer yaratacaktır. Bu süreç, tüketicilere zeki asistanlık hizmetlerinin, yeni tip araçlar tarafından sunulması ve yeni hizmet türlerinin ortaya çıkmasıyla sonuçlanacak süreçleri başlatacak, bu da yeni iş yapma biçimlerinin gelişmesine olanak sağlayacaktır.

Çalışma süreci iki aşamada değerlendirilebilir. Birincisi; ontoloji geliştirme süreci, diğeri ise geliştirilen ontolojinin işleme süreci. Bu çalışmada, ontoloji ve anlamsal web kavramlarını bilinen kavramlar ile ilişkilendirebilmenin, anlamsal web teknolojileriyle ilgili standartların farkında olmanın ve önemini kavramanın ve ontoloji geliştirme araçları ile metodolojileri tanımanın önemi incelendiği gibi Semantik Web altyapısı ile anlamsal sorgulama yapmayı sağlayan bir Kütüphane sistemi geliştirilmiştir.

Çalışmanın bilime kazandırdığı ve bu alanda yapılmış uygulamalara ek olarak getirdiği kapsamlardan bahsedecek olursak; ilk etapta Anlamsal ağ kavramının ortaya çıkış nedenleri, örnekleri, günümüzde Web 3.0'ın geldiği ve gelmesinin öngörüldüğü noktaların çeşitliliği, bu yönde yapılabilecek uygulamaların zaman, ürün kalitesi ve maliyet doğrultusunda kazandırdıklarını kavratmayı sağlamıştır. Daha sonrasında, uygulama tasarımı ve geliştirimi konularına gelindiğinde, varolan teknolojiler, bu altyapıyla yapılabilecekler ve uygulamanın gerçekleşmesi

durumunda elde edilen sonuçların verimliliği, yapılan bu çalışmada tam bir özgünlük ve tatminiyet sağlamıştır.

Ontoloji geliştirme süreci, ontoloji ile ilgili kavramların öğrenilmesi, ontoloji geliştirme, geliştirilen ontolojinin kullanımı gerekli altyapının oluşturulması gibi adımlardan oluşmaktadır. Ontoloji geliştirme aşamasında hissedilen en önemli eksiklik, bu alanda yapılmış pek çok uygulama, araştırma olmasına rağmen, bir anlamsal web uygulaması geliştirme aşamalarını yeni geliştiricilerin faydalanabileceği düzeyde pratik olarak açıklayan kaynakların eksikliği oldu. Ontoloji geliştirildikten sonra o ontolojinin nasıl işlenebileceği konusunda yapılan çalışma sayısı fazla olmasına rağmen ulaşılabilecek kaynak ve bilgi konusunda sıkıntılar yaşandı. Karşılaşılan pek çok kaynak ortamlarında, bu konuda çok fazla kişinin arayış içerisinde olduğu, ontolojiyi geliştirdikten sonra bu ontolojiyi nasıl kullanmaları gerektiğini bilmedikleri görüldü. Geliştirilen uygulamada da en çok zaman kaybı, basit de olsa, geliştirilen bir ontolojinin uygulama ortamında nasıl kullanılacağı konusunun araştırılması aşamasında yaşandı.

İleriki çalışmalarda bu sistem birleşik kütüphane sistemine dönüştürülmek üzere kullanılabilir. Örneğin, gerçek bir kütüphane sistemine gömülen yazılım, istenirse yapılabilecek çalışmalarla birlikte sipariş sistemi tutan ve kitap girişine izin veren bir sistem haline getirilebilir.

Son olarak, Semantik Kitap Arama Motoru projesinin genel açıdan özüne inmek ve ruhunu yansıtmak gerekirse belirtmek istenen esas hususlar bir sayfayla şöyle özetlenebilir:

Günümüz bilgisayar ve teknoloji atmosferinde yenilikler mekanik işlevsellikle, bilgi depolamayla ve bu bilgilerin doğru aktarımıyla ilgili olduğu kadar, kullanıcıya ne kalitede ulaştığıyla, ne kadar anlamsallık içinde olduğuyla ve kişiye göre nasıl davrandığıyla da yakından ilgilenir. Özellikle, bilgisayar ve internet teknolojilerinin farklı insanlarla daha derinlemesine iletişim kurabilmesi bakımından hızla gelişen bir dönemdeyiz. Eğer bilgisayar bilimini biraz da sosyal bilimlerle (örneğin

pazarlamayla) bağdaştırırsak, herkesin ihtiyaçlarının belli noktalarda özelleştiğini, ilgi alanlarının özgün seyretmesinden dolayı değişik şeylere ulaşmaya çalıştıkları görülür. Bu yüzden muhtelif pazarlar, hedefler ve ihtiyaçlar vardır ve amaç ihtiyaçları bulup karşılamaktır.

Bu felsefeden yola çıkarak, bilime sadece bilim olarak değil, “BilimSanat” olarak bakmamız gerektiğinin açık olduğu görülmektedir. Bilgisayar teknolojilerinin insanlara aradığı birçok şeyi, geniş ve artan veritabanı ile vermesi müthiş bir keşif. Çok güçlü bir altyapıdır. Bu keşif ve altyapının insanlara tam olarak aradığı şeyi vermesi ve anlamsal bir ilişki kurması ise zamanımızın filizlenen, potansiyeli heyecan verici olan bir alanıdır. İşte böyle özgün ve gelecek vaadeden bir çalışmayı bu yönünden geliştirmeye çalışırken, vakit kazanmak, konsantrasyonu korumak, motivasyonu arttırmak, mantıksal ilişkiyi daha anlamlı kurabilmek ve istekleri en özlü hallerde karşılamak adına Semantik Web Uygulamalarının teknoloji dünyasında fevkalade saygın ve önemli bir yeri olduğu kanısına varılmaktadır.

Bilim sanatla, yenilik gelişmeyle bir bütündür. Her yenilik başka bir gelişim, her gelişim başka bir yenilik olmalıdır.

KAYNAKLAR

1. Berners-Lee, T., Hendler, J., Lassila, O. , “The semantic web”, *Scientific American*, 184(5): 34-43 (2001).
2. Internet: Dave Beckett's Resource Description Framework “(RDF) Resource Guide” <http://www.ilrt.bristol.ac.uk/discovery/rdf/resources> (2005).
3. Internet: All Standards and Drafts “W3C Technical Reports and Publications” <http://www.w3.org/TR> (2010).
4. Celino, I., Valle, E.D., Cerizza, D., Turati, A., “Squiggle: a semantic search engine for indexing and retrieval of multimedia content”, *1st International Workshop on Semantic-enhanced Multimedia Presentation Systems* , Milano, 7-11 (2006).
5. Cohen, S., Mamou, J., Kanza, Y., Sagiv, Y., “Xsearch: a semantic search engine for xml,” *VLDB Conference*, Berlin, 45–56 (2003).
6. Imielinski, T., Signorini, A., “If you ask nicely, I will answer: semantic search and today’s search engines”, *Scientific American*, 10:5-10, (2006).
7. Internet: English in global multimedia “A Virtual Empire?” <http://pietergroenendijk.files.wordpress.com/2010/01/web123.gif> (2010).
8. Daconta, Michael C., Obrst, Leo J., and Smith, Kevin T., “The semantic web: a guide to the future of XML, web services, and knowledge management, 1st ed.”, *Wiley Publishing Inc.*, New York, 10-15, (2003).
9. Internet: 3C semantic web activity, “World Wide Web Consortium” <http://www.w3.org/2001/sw/>, (2007).
10. Internet: Tools: Semantic Web, “Semantic Web Development Tools”, <http://www.w3.org/2001/sw/wiki/Tools>, (2007)
11. Baader, F., Horrocks, I., Sattler, U., “Description logics as ontology languages for the semantic web”, *BaHS03 Conference*, Germany, 228-248, (2003).
12. Internet: Bilgisayar Kavramları “Semantic Stack” <http://www.bilgisayarkavramlari.com/wp-content/uploads/2009/06/sematicstack.png> (2010).
13. Andersson, B., Bergholtz, M., Edirisuriya, A., Ilayperuma, T., Johannesson, P., Grégoire, B., et al. “Towards a common ontology for business models”, *Proceedings of the Open INTEROP Workshop On Enterprise Modelling and Ontologies for Interoperability (EMOI-INTEROP)*, Luxembourg, 10-15, (2006).

14. Farrar, S., Lewis, W. & Langendoen, T. (2002). "A common ontology for linguistic concepts", *In Proceedings of the Knowledge Technologies Conference, Seattle*, 10-15, (2002).
15. Internet: Akyokuş, S., "Anlamsal Web Sunu", http://www.edirnevdb.gov.tr/kultur/ppt/anlamsal_web_rdf_dc_owl.ppt#326,11 (2010).
16. Internet: Vikipedi Özgür Ansiklopedi "Anlamsal Web" http://tr.wikipedia.org/wiki/Anlamsal_a%C4%9F, (2010).
17. Kurtel, K., "Anlamsal web, ege akademik bakış", *Ege Academic Review*, 8 (1): 205-213, (2008).
18. Internet: Doğdu, E., "Semantik web", edogdu.etu.edu.tr/course/bil554/lectures/21-semantic-web.pdf, (2010).
19. Internet: RDF "Calendar Notes" <http://www.ilrt.bris.ac.uk/discovery/2001/02/calendar/> (2001).
20. Internet: Adobe XMP "Adding Intelligence to Media" <http://www.adobe.com/products/xmp/main.html> (2011).
21. Internet: PhotoStuff "An image annotation tool for the Semantic Web" <http://www.mindswap.org/2003/PhotoStuff/> (2010).
22. Internet: SMORE "About Smore" <http://www.mindswap.org/2005/SMORE/> (2010).
23. Internet: Piggy Bank- SIMILE, "About Piggy Bank" <http://simile.mit.edu/piggy-bank/> (2008).
24. Internet: Hayloft, SIMILE "About Hayloft" <http://simile.mit.edu/hayloft/index.html> (2008).
25. Internet: FOAF "About FOAF" <http://www.foaf-project.org/> (2008).
26. Internet: Last FM "About Last Fm" <http://www.last.fm/> (2010).
27. Internet: The HERA Research Program "About HERA" <http://www.wis.win.tue.nl/~hera> (2010).
28. Internet: ESW Wiki "Semantic Web Tools" <http://esw.w3.org/topic/SemanticWebTools> (2010).
29. Bozsak, E., Ehrig, M., Handschuh, S., "Kaon – towards a large scale semantic web", *Proc. Of EC-Web 2002, LNCS*, Provençe, 304-313, (2002).

30. Internet: Webopedia “What is SGML”
<http://www.webopedia.com/TERM/S/SGML.html> (2011).
31. Buraga, S. C., Rusu, T., “Search Semi-Structured Data on Web”, *The Seventh International Symposium on Automatic Control and Computer Science Proceedings*, Iasi, 2-5, 2001.
32. Deutsch, A., Fernandez, M., Florescu, D., Levy, A., Suciu, D., “A query language for xml”, *Submission to the World Wide Web Consortium*, 4:66-88, (1998).
33. Ding Ying, Fensel Dieter, Klein Michel, and Omelayenko Borys, “The semantic web: yet another hip?”, *Data and Knowledge Engineering*, 41:205-227, (2002).
34. Deconta, M.C., Obrst, J.L.; Smith T.K., “The semantic web 2nd ed.”, *Wiley*, NewYork, 58 (2003).
35. Internet: Wikipedia “Kaynak Tanımlama Çerçevesi”
http://tr.wikipedia.org/wiki/Kaynak_Tan%C4%B1mlama_%C3%87er%C3%A7evesi (2011).
36. Internet: RDF “Kaynak Tanımlama Çerçevesi”,
http://www.emo.org.tr/ekler/c2f61d37674400a_ek.pdf (2009).
37. Hongbing, W., Zhexue, J. H., Yuzhong, Q., Junyuan, X., “Web services: problems and future directions”, *Journal of Web Semantics, Web Semantics: Science, Services and Agents on the World Wide Web*, 9(2): 83-244, (2004).
38. Internet: Semantic Web “Introduction, Benefits and Detailed Example”
<http://encyclopedia.jrank.org/articles/pages/6898/Semantic-Web.html#ixzz0rBp4uMZh> (2010).
39. Internet: Türkyılmaz, İ., “Semantik Web Teknolojileri”, *Akademik Bilişim 2008 Çanakkale Onsekiz Mart Üniversitesi, Çanakkale*,
http://ab.org.tr/ab08/kitap/Bildiriler/90,1_40_AB08.pdf (2008)
40. Internet: Ontology “What is an ontology?” <http://www-ksl.stanford.edu/kst/what-is-an-ontology.html> (2011).
41. Internet: What is an ontology and why we need it? “Ontology Development 101: A Guide to Creating Your First Ontology”
http://protege.stanford.edu/publications/ontology_development/ontology101-noy-mcguinness.html (2010).

42. Ullrich, C., “Description of an instructional ontology and its application in web services for education”, *Proceedings of Workshop on Applications of Semantic Web Technologies for E-learning SW-EL'04*, Dublin, 17-23, (2004).
43. Internet: Gruber, T. “What is an ontology?”, <http://www.wksl.stanford.edu/kst/what-is-an-ontology.html>, (2007).
44. Internet: W3Chima “Frequently Asked Questions on Ontology Technology” <http://bbs.w3china.org/ispbbs.asp?boardid=2&id=23255&star=1> (2010).
45. Internet: What you need, when you need it “Anlamsal Ağ (Semantik Web) Nedir?” [http://www.ahmetalisuzen.com/aas/post/2009/08/03/Anlamsal-Ag\(Semantic-WEB\)-Nedir.aspx](http://www.ahmetalisuzen.com/aas/post/2009/08/03/Anlamsal-Ag(Semantic-WEB)-Nedir.aspx) (2010).
46. Internet: Kurtel, K., “Web’in geleceği: anlamsal web”, http://eab.ege.edu.tr/pdf/8_1/C8-S1-M11.pdf, (2008).
47. Internet: Anlamsal Web “Göksel Üçer, OWL Belgesi Hazırlama” http://ndn.netsis.com.tr/GokselUCER/Doktora/OWL_Yapisi.ppt#264,8,Namespaces (2010).
48. Internet: Wikipedia “Ontology (Information Science)” [http://en.wikipedia.org/wiki/Ontology_\(computer_science\)](http://en.wikipedia.org/wiki/Ontology_(computer_science)) (2010).
49. Internet: Semantic Web “Ontology” <http://semanticweb.org/wiki/Ontology> (2010).
50. Internet: Stı-Innsbruk “OntoWeb” <http://www.sti-innsbruck.at/fileadmin/documents/deliverables/Ontoweb/D1.5.pdf> (2010).
51. Ding Y., Fensel D., “Division of mathematics and computer science, ontology library systems: the key to successful ontology re-use”, *Proceedings of SWWS, Amsterdam*, 93-112, (2001).
52. Xiaomeng, S., Lars, I, “A comparative study of ontology languages and tools”, *Advanced Information System Engineering, 14th International Conference, CAiSE 2002*, Trondheim, 761-765, (2002).
53. Internet: Jena Semantic Web “Jena: A Semantic Web Framework” <http://www.hpl.hp.com/semweb/jena.htm> (2010).
54. Keskin, Ö., Sezer E., “Kaynak kod sorgulamada ontoloji kullanımı”, *4. Ulusal Yazılım Mühendisliği Sempozyumu - UYMS'09*, İstanbul, 5-10, (2009).
55. Internet: Açık Kaynak Döküman Projesi “Katmanlı Mimari Nedir? Faydaları Nelerdir?” <http://www.bilgininadresi.net/> (2009).

56. Internet: Herman, I. “Web ontology language (owl)”, <http://www.w3.org/2004/OWL/>, (2004).
57. Internet: Resource DescriptionFramework RDF “Resource Description Framework (RDF) Model and Syntax Specification” <http://www.w3.org/TR/1999/REC-rdf-syntax-19990222/#model> (1999).
58. Internet: Semantic Web “SemWeb – DotNet” <http://semanticweb.org/wiki/SemWeb-DotNet> (2008).
59. Internet: SemWeb.NET “Semantic Web/RDF Library for C#/.NET” <http://razor.occams.info/code/semweb/> (2009).
60. Internet: SemWeb Documentation “SemWeb:Docs” <http://razor.occams.info/code/semweb/semweb-current/doc/overview.html> (2010).
61. Internet: MSDN Documents “XmlDocument Class” <http://msdn.microsoft.com/en-us/library/system.xml.xmldocument.aspx> (2011).
62. Internet: SemWeb.RdfXmlReader “RdfXmlReader Class” <http://razor.occams.info/code/semweb/semweb-current/apidocs/SemWeb/RdfXmlReader.html> (2010).
63. Internet: C# ve ASP.NET Makale Portalı “LINQ Nedir?” <http://www.ibrahimmarac.com/post/LINQ-Nedir.aspx> (2010).

ÖZGEÇMİŞ

Kişisel Bilgiler

Soyadı, adı : KAZANDIR, Zehra Burcu
Uyruğu : T.C.
Doğum tarihi ve yeri : 26.08.1986 Ankara
Medeni hali : Bekâr
Telefon : 0 (506) 856 56 09
e-mail : burcukazandir@gmail.com

Eğitim

Derece	Eğitim Birimi	Mezuniyet tarihi
Yüksek lisans	Gazi Üniversitesi /Yönetim Bilişim Sistemleri	(Devam)
Lisans	Başkent Üniversitesi/ Bilgisayar Mühendisliği	2008
Lise	Hacı Ömer Tarman Anadolu Lisesi	2004

İş Deneyimi

Yıl Yer Görev

2008-... TAI- Türk Havacılık ve Uzay Sanayii A.Ş. - Yazılım Mühendisi
2007 ASELSAN A.Ş. (Staj)
2007 HAVELSAN A.Ş. (Staj)
2006 MODSİMMER A.Ş. (Staj)

Yabancı Dil

İngilizce İleri Düzey
İspanyolca Başlangıç Düzeyi

Hobiler

Piyano, Spor, Bilgisayar teknolojileri.

