

Anlamsal Web Servisleri Temelinde Örnek Bir Servis Tanımı

Yasemin Yüksek

Ege Üniversitesi, Bilgisayar Mühendisliği Bölümü, İzmir
yasemin.yuksekk@ege.edu.tr

Özet: Şu anda kullanılmakta olan web servis teknolojileri anlamsal birlikte işlerliği sağlamamaktadır. Anlamsal birlikte işlerliği sağlamak için web servis yeteneklerinin servis ontolojileri kullanılarak tanımlanması ve bu ontolojilere gereksinim duyulduğunda dinamik olarak erişimin sağlanması gerekmektedir. Bu çalışmada web servis tanımları ve web servis yeteneklerinin ontolojilerde nasıl saklandığı anlatılmıştır. Örnek çalışma olarak heterojen ortamda birlikte işlerliği sağlayan OWL servis ontolojisi tanımlarından oluşan bir sistem tasarlanmıştır. Servislere nasıl erişileceğini belirten OWL-S servis zemin(grounding) ontolojisi için WSDL ile tanımlanan bir arayüz kullanılmıştır.

Anahtar Kelimeler: Anlamsal Web Servisi, OWL-S, WSDL.

Abstract: Current web service technologies does not provide semantic interoperability. To provide this, the abilities of web services have to be defined by using service ontology and these ontologies have to be access dynamically when it is needed. This paper mentions how web service definitions and abilities of web services can be stored in ontologies. As a study case, a system that supports to define OWL service ontologies, was designed on the verge of defining services that cooperate in heterogeneous environments. To define OWL-S service grounding ontology that provides to access web services, an interface that developed by WSDL, uses.

Keywords: Semantic Web Service, OWL-S, WSDL.

1. Giriş

Web servisleri, farklı programlama dilleri kullanılarak yazılmış ve farklı platformlarda bulunan yazılım bileşenlerinin web standartları kullanılarak web ortamında yayınlanmasına, aranıp bulunabilmesine ve istenildiğinde çağrılarak biçimli ve evrensel erişimine izin veren teknolojilerdir[2].

Şu anda kullanılmakta olan web servis teknolojilerinin en büyük eksikliği, otomatik keşfetme ve birleşme özelliğinin olmamasıdır. Bununla birlikte web servis yetenekleri statik biçimde tanımlanması nedeniyle servislerin özellikleri, yetenekleri, arayüzleri ve etkileri makineler tarafından anlaşılabilir yapıda değildir.

Web ortamında bilgi sistemlerinde saklanan bilginin anlamsal seviyede tanımlanması, aranması ve paylaşımı gerekmektedir. Bunun için son yıllarda bilgiler arasında anlamsal ilişkilerin kurulması ve bu anlamsal ilişkinin makinelerin arasında anlaşılması için Anlamsal Web teknolojileri geliştirilmiştir.

Web tabanlı bilgi işlemlerinin paylaşımında ve yeniden kullanımda ontolojilerin önemli bir rolü vardır. Ontoloji, paylaşımlı kavramsallaştırmanın biçimsel ve açık şekilde sunumudur. Kavramsallaştırma, belirli bir alan ile kısıtlı kavramlar üzerinde düşüncelerin soyut modellenmesidir. Ontolojinin diğer amaçlarından birisi de farklı uygulamalarda yeniden kullanılabilir ve paylaşılabilir bir modelin oluşturulmasıdır[3].

Web ortamındaki gelişmelerden servis süreçlerinin bütünleştirilme gereksinimi sonucunda anlamsal web tanımlamaları kullanılarak web servislerinin yetenekleri anlamsal olarak artırılmış. Anlamsal web ve web servis teknolojilerinin birleştiği noktada anlamsal web servisleri, veri ve bilginin otomatik olarak işlenmesi ve platformlar arasında gönderilemesini sağlamaktadır.

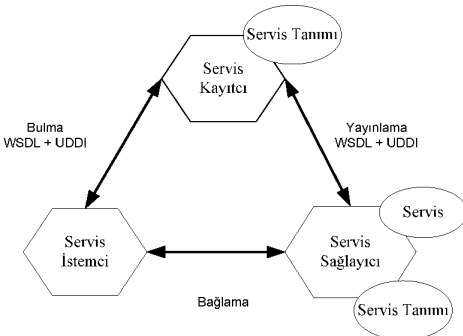
Bu çalışmada web servis teknolojileri ve web servis yeteneklerinin ontolojilerde nasıl saklandığı anlatılmıştır. İkinci ve üçüncü bölümlerde web servisleri ve anlamsal web servisleri hakkında bilgiler verilmiştir. Dördüncü bölümde de heterojen ortamda birlikte işlerliği sağlayan servislerin, OWL servis ontolojisi tanımlama işlemlerini gerçekleştiren bir sistem tasarlanmış. Son bölümde ise gerçekleştirilen sistem hakkında sonuçlar değerlendirilmiştir.

2. Web Servisleri

2.1 Web Servis Mimarisi

Web servisi web ortamında belirli görev veya görevleri yerine getiren işlemleri tanımlamak için bir arayüz oluşturmaktadır. Bir web servisi, WSDL(Web Service Description Language) kullanılarak tanımlanır. Servis yayınlama, bulma, bağlanma ve çağırma işlemlerinin gerçekleştirmek için SOAP(Simple Object Access Protocol) standardı kullanılmaktadır.

Web servisleri için ortak mimari şekil 1’de resimlenmiştir[4].



Şekil 1. Web Servis Mimarisi

Servis istemcisi(“ıServzce Requester”) servisleri çağırır, servis sağlayıcı(“^ervz’ce Provider”) istemcinin isteklerini cevaplamaktadır. Servis kayıtçısında(Service Registry) servis sağlayıcı tarafından yayınlanan servis tanımları ilan edilmekte ve yayınlanmaktadır. Servis sağlayıcı servis kayıtçısında servisleri tarayarak bu servisler hakkında bilgiler elde eder.

2.1.1 Web Servis Tanımlama Dili

Web servis tanımlama dili(WSDL), servisin işlevini, servise nasıl erişileceğini tanımlamak için kullanılan XML tabanlı dildir. WSDL belgesi, servis tanımları için aşağıdaki elemanları içermektedir.

Tipler: Veri tiplerini tanımlar.

Mesaj: Servis sağlayıcı ve servis istemcisi arasındaki iletişimi kurabilmek için kullanılan mesajları tanımlar.

İşlem: Servis sağlayıcı ve servis istemcisi arasında mesaj değişimi işlemler kullanılarak tanımlanır.

Port Tipi: Web servisinin içerdiği işlemleri tanımlar.

Bağlayıcı: Mesajlar için kullanılan veri formatlarını tanımlar.

Port: Bağlayıcı ve servis web adresinden oluşan servis noktasını tanımlar.

Servis: Bütün ilişkili portların topluluğudur.

2.1.2 Evrensel Tanımlama

Keşfetme ve Bütünleştirme

Servis sağlayıcıların oluşturdukları web servislerin diğer servis istemcileri tarafından aranıp bulunabilmesi gerekebilir. Bu nedenle web servislerine ait bilgilerin bulunduğu, istemci tarafından servise nasıl erişileceğini, nerde bulabileceğini ve de en önemlisi nasıl kullanacağını öğrenebilecekleri bir yapıya gereksinimi vardır.

UDDI(Universal Description Discovery and Integration)[8] bu gereksinimleri karşılamak

için kullanılan yöntemlerden ilkidir. UDDI' ı oluşturan kelimeler aslında yaptığı işi özetliyor. Universal- evrensel bir servis, description- web servislerin tanımlanması, discovery- diğer web servislerin keşfedilmesi, integration-global servisler ile entegre olduğu göstermektedir.

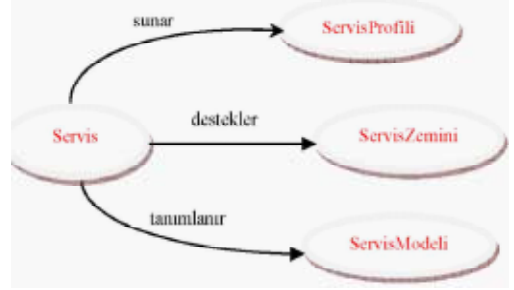
Fakat günümüzde kullanılan UDDI kayıtçısının bazı yetersiz özellikleri bulunmaktadır. UDDI kayıtçısı veri şemaları, üstverileri veya veri modellerini saklayabilecek ve yönetebilecek özelliklere sahip değildir. Bu özelliklerin eksikliği nedeniyle kayıtlanan bilgiler arasında ilişkiler kurulamaz. Bu ilişkilerin kurulması için aşağıdaki koşulların gerçekleştirilmesi gerekmektedir[1].

- Servislerin anlamsal tanımları UDDI kayıtçısında depolanmalıdır.
- Servis kullanıcıları ve servis sağlayıcılar kendi aralarında anlamsal tanımlardaki standart sunum belirtilerinde anlaşmaları gerekir.
- UDDI kayıtçısı, servis tanımları ve kullanıcı istek tanımları arasında anlamsal eşleme ve çıkarsama yeteneğine sahip olmalıdır.

3.Anlamsal Web Servisleri

Anlamsal Web, web servis tanımlarında karşılaşılan anlamsallık problemini çözmektedir. RDF(Resource Description Framework), OWL(Web Ontology Language) gibi Anlamsal Web'de ontoloji tanımlama dilleri kullanılır. OWL-S(OWL for Services) heterojen ortamda birlikte çalışabilen servislerin yayınlanması, servislerin seçimi ve servisler arasında otomatik olarak mesaj değişim desteğini veren anlamsal web servislerini oluşturulan OWL-S ontolojisidir[5].

Aşağıdaki Şekil 3.1' de OWL-S ontolojisi hakkında üç çeşit bilgi sağlamaktadır.



Şekil 3. OWL-S Ontolojisi üst seviyesi[9]

Birlikte işlerliği ve tümleşimi sağlamak amacıyla web servislerinin yeteneklerinin tanımlanması ve servis çalıştırma sürecine yönelik OWL-S ontolojisi kullanılır. OWL-S ontoloji tanımlarında değişik servis özellikleri (input-giriş, outputs'-çıkış, preconditions-önkoşullar, e^cts-sonuçlar (IOPE)) kullanılır. OWL-S ile tanımlanan web servis yapısı üç parçadan oluşmaktadır. Bu parçalar servis profili, servis modeli ve servis zemini. Servis profili(Service Profile), servis bulma ve servis ilan etme(reklam yapılması) işlemlerini sağlar, işlem modeli(Process Model) servis işlemlerinin ayrıntılı tanımlarını yapar ve servis zemini(Service Grounding) mesajlar aracılığıyla servisler arasında birlikte işlerliği sağlar[9].

4. Örnek Bir Anlamsal Web Servisi Geliştirimi

4.1Anlamsal Servis Tanımlama Süreci

OWL-S ontolojisi, servisinin giriş, sonuç, önkoşul ve sonuç bilgilerini içerir. Böylelikle web servis işlem metodları arasındaki kontrolünün ve veri akışının sağlanmasıyla web servisinin yürütme işlemi gerçekleşir.

Anlamsal web servisleri, tanımlanan alan ontolojilerindeki bilgileri giriş verileri olarak kullanarak istenilen sonuç verilerinin geri döndürmektedir.

Her web servisi, uzak istemcilerin başvuruda bulunduğu çeşitli işlem çağırımlarını barındırmaktadır. Bu bölümde örnek olarak gösterilecek servislerin atomik işlemleri modellemiştir.

Atomik işlem, bir nesnenin bir özelliği sadece bir işlem tarafından değiştirilir. Atomik işlemin çalıştırılma sürecinde giriş değerlerine göre bize sonuç değerini döndürmektedir.

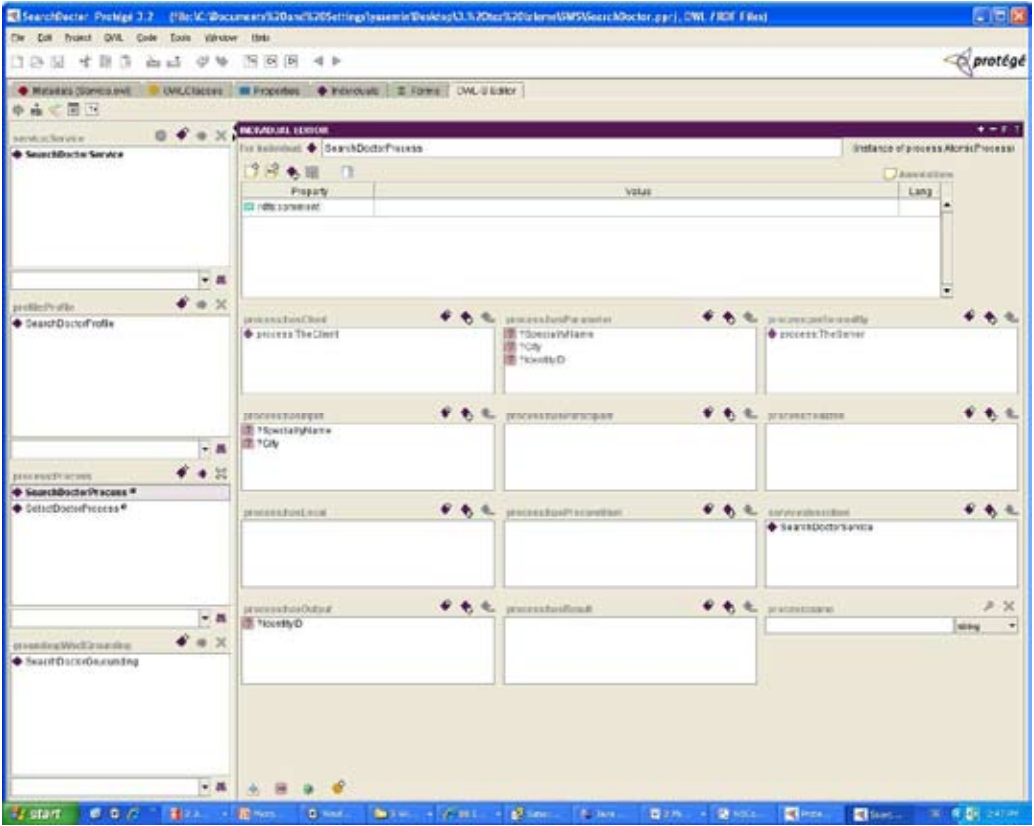
Bu çalışma için örnek doktor arama servisi için doktor arama atomik işlemin giriş ve çıkış verileri Şekil 4.1' de gösterilmiştir.

Servis ontolojilerini oluşturulması için, Protege [6] ontoloji geliştirme editörünün OWL-S ek-

lentisi kullanılmıştır. Şekil 4.2' de DoctorSearchService servisine ait ekran görüntü verilmiştir.



Sekil 4.1 Doctor_Search atomik işlemin giriş ve çıkış verileri



Sekil 4.2 SearchDoctorService servisinde tanımlanan SearchDoctorProcess işleminin giriş-çıkış verilerinin tanımlandığı ekran görüntüsü

OWL-S ile tanımlanan web servis yapısı parçasından birisi olan servis zemini, mesajlar aracılığıyla servisler arasında birlikte işlerliği WSDL belgesi kullanarak sağlamaktadır. WSDL belgesindeki her bir

işlem için, bu işleme OWL-S atomik işlemde karşılık gelecek metod tanımları yapılır. SearchDoctorService servisinde SearchdoctorGrounding servis zemini için SearchDoctorGrounding .wsdl[7] belgesi tanımlanmıştır.

WSDL belgesinin temel elemanları:

Tip kısmı;

```
<wsdl:types>
<xsd:schema
targetNamespace="http://efe.ege.edu.tr/~yukse
k/Templates/SearchDoctorGrounding">
<xsd:element name="GetDoctorList">
<xsd:complexType>
<xsd:sequence>
<xsd:element
name="SpecialityName" type="xsd:string"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
.....
```

Geri döndürülecek değer için eleman tanımı;

```
<xsd:element
name="GetDoctorListResponse">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="IdentityID"
type="xsd:string"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
```

SOAP mesajları içine konan ve SOAP mesajları ile geri dönen parametreler;

```
<wsdl:message
name="GetDoctorListSOAPIn">
<wsdl:part element="tns:GetDoctorList"
name="parameters"/>
</wsdl:message>
<wsdl:message
name="GetDoctorListSOAPOut">
<wsdl:part
element="tns:GetDoctorListResponse"
name="parameters"/>
</wsdl:message>
```

Web servis işlemleri ve SOAP mesajlarını içeren bölüm;

```
<wsdl:portType
name="SearchDoctorGrounding">
```

```
<wsdl:operation name="GetDoctorList">
<wsdl:input
message="tns:GetDoctorListSOAPIn"/>
<wsdl:output
message="tns:GetDoctorListSOAPOut"/>
</wsdl:operation>
```

Mesajlar için kullanılan veri tiplerinin tanımlandığı bölüm;

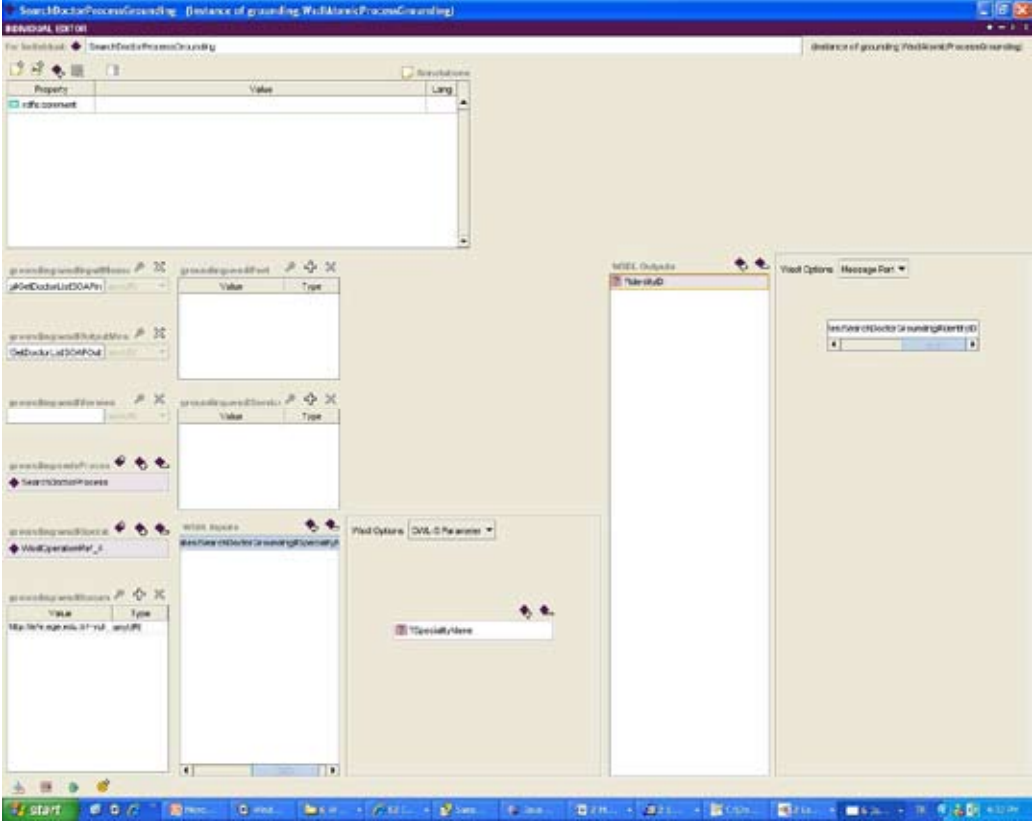
```
<wsdl:binding
name="SearchDoctorGroundingSOAP"
type="tns:SearchDoctorGrounding">
<soap:binding style="document"
transport="http://schemas.xmlsoap.org/soap/http"/>
<wsdl:operation name="GetDoctorList">
<soap:operation
soapAction="http://efe.ege.edu.tr/~yukse
k/Templates/SearchDoctorGrounding/GetDoctorList"/>
<wsdl:input>
<soap:body use="literal"/>
</wsdl:input>
<wsdl:output>
<soap:body use="literal"/>
</wsdl:output>
</wsdl:operation>
</wsdl:binding>
```

Bütün ilişkili portların tanımladığı kısım;

```
<wsdl:service
name="SearchDoctorGrounding">
<wsdl:port
binding="tns:SearchDoctorGroundingSOAP"
name="SearchDoctorGroundingSOAP">
<soap:address
location="http://efe.ege.edu.tr/~yukse
k/Templates/">
</wsdl:port>
</wsdl:service>
```

tanımları örnek olarak verilmiştir.

Şekil 4.3' de resimlendiği gibi servis ontoloji tanımında WSDLAtomicProcessGrounding kısmı ile yukarıda örnek olarak verilen WSDL belgesinin bütünleştirilmiş görüntü verilmiştir.



Sekil 4.3 SearchDoctorGrounding servis zemin ontolojisinin gösterimi

5. Sonuç ve Yorumlar

Yeni teknolojilerden birisi olan anlamsal web ile web servislerinin kesiştiği noktada anlamsal web servislerine gereksinim duyulmaktadır. Servislerinin anlamsal arayüzleri ve yetenek tanımları servis ontolojisi kullanılarak oluşturulmaktadır. Bu ontoloji servis sağlayıcıları tarafından servis kayıtcısında saklanması ve ilan edilmesi için kullanılmaktadır. Ayrıca bu ontoloji servis istemcisi tarafından anlamsal web servislerinin çalıştırılmasında kullanılmaktadır.

Bu çalışma kapsamında yukarıda örneklenen tek bir servisin tasarımı ile ilgilidir. Karmaşık bir işlemi gerçekleştirmek için birden fazla ser-

visi ihtiyaç duyulabilir. Bu çalışmanın devamı olarak karmaşık anlamsal web servislerinin tasarımı hedeflenmektedir.

6. Kısaltmalar

Kısaltma: Açıklama:

OWL	Web Ontoloji Dili
OWL-S	Servisler için Web Ontoloji Dili
SOAP	Basit Nesne Erişim Protokolü
SWS	Anlamsal Web Servis
UDDI	Evrensel Tanımlama
Keşfetme ve Bütünleştirme	
WSDL	Web Servisleri Tanımlama Dili
XML	Genişletilebilir Biçimleme Dili

7. Kaynaklar

- [1]. Paolucci, M., Kawamura, T., Payne, T., R., Sycara R., 2002, "Importing the Semantic Web in UDDI", In Proceedings of E-Services and the Semantic Web Workshop, 2002.
- [2]. Dutta, B., Semantic Web Service: A study of Existing Tehnologies, Tools and Projects", DESIDOC, 2008.
- [3]. T. Berners-Lee, J. Hendler, and O. Lassila "The Semantic Web", Scientific American, vol. 184, no. 5, pp. 34--43, 2001.
- [4]. Gottschalk, K., Graham, S., Kreger, H. and Snell, J., "Introduction to Web Services Architecture", New Development in Web Services and E-commerce, 2002.
- [5]. Srinivasan, N, Paolucci, M and Sycara, K., 2004, CODE: A Development Environment for OWL-S Web Services. Demo paper in 3rd International Semantic Web Conference.
- [6]. Protege Ontology Editor, <http://protege.stanford.edu/>
- [7]. SearchDoctorGrounding tanımı, <http://efe.ege.edu.tr/~yukse/ Templates/ SearchDoctorGrounding .wsdl>
- [8]. UDDI Spec TC, 2002 <http://www.uddi.org/pubs/uddi v3 .htm>
- [9] OWL-S Coalition, 2004, <http://www.w3.org/Submission/OWL-S/>