

KONUMSAL VERİ ALTYAPILARI İÇİN KATALOG SERVİSİ STANDARTLARININ DEĞERLENDİRİLMESİ

H. Akıncı¹, Ç. Cömert²

¹ Ondokuz Mayıs Üniversitesi, Jeodezi ve Fotogrametri Müh.Bölümü, Kartografya Anabilim Dalı, Samsun hakinci@omu.edu.tr
² Karadeniz Teknik Üniversitesi, Jeodezi ve Fotogrametri Müh.Bölümü, Kartografya Anabilim Dalı, Trabzon ccomert@ktu.edu.tr

ÖZET

Konumsal Veri Altyapıları (KVA), katılımcı eşler arasında birlikte işlerliği sağlayarak hızlı, ekonomik ve kaliteli servis sağlamaya olanak tanır. KVA'ları gerçekleştirmek için şu an yaygın olan eğilim, "Servis Yönelimli Mimari" (SyM) dir. Web servisleri, SyM'yi gerçekleştirmenin en iyi ve şu anki en popüler yolu olarak kabul edilmektedir. Bununla birlikte, bir KVA'nın Web servisleri ile başarılı bir gerçekleştirimi için üzerinde durulması gereken birçok sorun mevcuttur. Bu sorunlardan biride, katalog servislerinin gerçekleştirilmesi ile ilgilidir. Katalog servisleri için konumsal ve konumsal olmayan bilgi toplumu tarafından geliştirilen çeşitli standartlar ve belirtiler mevcuttur. Buradaki sorun, bu standartların nasıl gerçekleştirileceği ve Ulusal KVA (UKVA) gibi geniş ölçekli bir KVA da kataloglar arasında birlikte işlerliğin nasıl sağlanacağıdır. Bu çalışmanın amacı, katalog servislerine özel sorunları tespit etmek ve bunlar için çözüm önerileri getirmektir.

Anahtar Sözcükler: Konumsal Veri Altyapısı, Servis Yönelimli Mimari, Web Servisleri, Katalog Servisleri

ABSTRACT

EVALUATING CATALOGUE SERVICE STANDARDS FOR SPATIAL DATA INFRASTRUCTURES

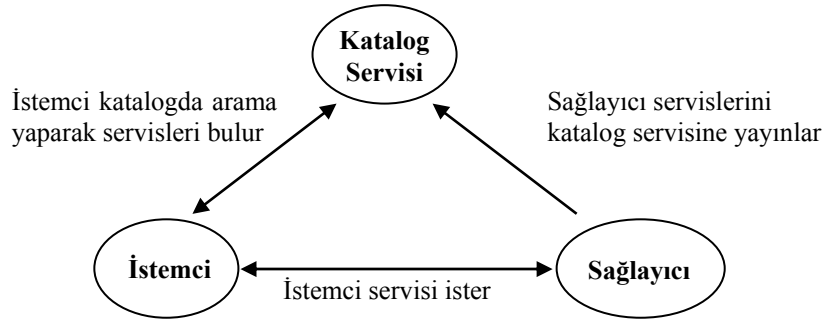
Spatial Data Infrastructures (SDI) ensure rapid, economic, and high quality service provision enabling interoperability among its participating partners. The current trend for implementing SDIs is to employ a Services Oriented Architecture (SOA). Web Services has been accepted as one of the best and currently the most popular way of implementing SOA. However, there are many issues to be addressed for a successful implementation of a SDI with the Web Services. One of these issues is related to the implementation of catalogue services. There are a number of standards and specifications for catalogue services for both Geospatial and non-Geospatial communities. The issue here is how to implement these standards and provide interoperability among them within a large scale SDI environment like National SDI (NSDI). The aim of this paper is to identify these catalogue-services-specific issues and propose solutions for them.

Keywords: Spatial Data Infrastructure, Services Oriented Architecture, Web Services, Catalogue Services

1. GİRİŞ

Konumsal Veri Altyapıları (KVA), konumsal veri ile iş yapan bütün kesimler arasında "birlikte işlerliği" sağlayan ve vatandaşlar dahil ilgililere, gereksinim duydukları veri ve servislere anında erişim ve kullanım olanağı tanıyan bir altyapıdır. KVA'lar bir birlikte işlerlik altyapısıdır. Birlikte işlerlik, genel olarak farklı dil ya da kavramlar kullanan uygulamaların birbirleri ile konuşabilmesi olarak tanımlanabilir. Birlikte işlerliği sağlamak için geliştirilen ve şu an oldukça popüler ve yaygın olan yazılım mimarisi, "Servis Yönelimli Mimari" (Service Oriented Architecture) ya da kısaca SyM (SOA) olarak adlandırılmaktadır. Web servisleri, SyM'yi gerçekleştirmenin en iyi ve şu anki en popüler yolu olarak kabul edilmektedir. Ancak KVA'ların, Web servisleri ile gerçekleştirilebilmeleri Dünya genelinde henüz başarılı olmuş bir görev değildir. Çünkü böyle bir gerçekleştirimin, teknik birlikte işlerlik altyapısını tanımlayan, üzerinde anlaşma sağlanmış bir çerçeve mevcut değildir.

Üst düzey bir görüşle, bir Web servisi belirli bir görevi gerçekleştirmek için internet üzerinden çağrılabilen bir kod parçası olarak tanımlanabilir. Bir Web servisleri ortamı, servis sağlayıcılarının sahip oldukları Web servislerini bir katalog servisi aracılığıyla yayınladıkları ve istemcilerin katalogdan servisleri bulup, uygulamalarını gerçekleştirmek için onları sağlayıcılardan istedikleri bir ortam olarak kavramsallaştırılabilir (Şekil 1). Katalog servisleri, SyM'nin kilit bileşeni durumundadır. Çünkü SyM'de servis sağlayıcılar Web servislerini katalog servisleri üzerinden kullanıma sunar, istemciler de aradıkları özellikteki Web servislerini bu sayede tespit ederler. Diğer bir anlatımla katalog servisi, sağlayıcıların Web servislerini katalog servisine kaydetmelerine ve istemcilerin istenen özellikleri sağlayan Web servislerini aramalarına olanak tanır. Burada söz konusu olan iki işlem, sırasıyla, SyM'nin "yayınla" (publish) ve "bul" (find) işlemleridir. Ancak mevcut Web servisleri uygulamalarında katalog bileşeninin genellikle eksik olduğu görülmektedir. Diğer bir anlatımla, istemcilerin aradıkları özellikteki servisleri bir şekilde "bulmuş oldukları" kabul edilmektedir. Oysa SyM'nin potansiyelleri ve sorunları bakımından, gerek teorik ve gerekse pratik olarak değerlendirilebilmesinde, onun en temel bileşeni olan katalog boyutundaki değerlendirmeler son derece kritik bir rol oynayacaktır.



Şekil 1. Servis yönelimli mimarinin bileşenleri (Vinoski, 2002).

Web servisleri vizyonunda kritik nokta, istemcilerin aradıkları özellikleri sağlayan servisleri bulmaları ile ilgilidir. Aranılan servis ya da servisler bulunduğundan sonra geriye yalnızca servisin icrası ya da birden çok servis olması durumunda, servislerin belirli bir düzenleme ile icrası kalmış olacaktır. Özellikle Ulusal Konumsal Veri Altyapısı (UKVA) gibi geniş ölçekli bir KVA da çok sayıda servis ve bunların sağlayıcıları söz konusu olacaktır. Bu durumda bir istemci, gereksinimlerini karşılayacak özelliklerdeki bir ya da daha çok servisi nasıl bulacaktır? Bunun ne gibi boyutları vardır? Bu bağlamda katalogların önemi nedir? gibi sorular bu çalışmada ele alınacak konunun özünü teşkil edecektir.

Katalog servisleri için konumsal ve konumsal olmayan bilgi toplumu tarafından çeşitli standartlar ve belirtiler geliştirilmiştir. ebXML (electronic business XML) Registry and UDDI (Universal Description Discovery, and Integration) konumsal olmayan bilgi toplumu tarafından geliştirilen katalog servisi standartlarıdır. CSW (Catalogue Service Web) ise konumsal bilgi toplumu için OGC Konsorsiyumu tarafından geliştirilen katalog servisi belirtilimidir. Bugün piyasada sözkonusu standartları ve belirtileri gerçekleştiren çeşitli yazılım ürünleri mevcuttur. Ancak bunların hangilerinin yaygın kabul göreceği, farklı standartlar arasında uyumun nasıl sağlanacağı henüz belirsizdir. Bu nedenle KVA gerçekleştirenlerin de nasıl bir yol izleyeceği belirsizdir. Buradaki bir diğer sorun da, bu standartların nasıl gerçekleştirileceği ve UKVA gibi geniş ölçekli bir KVA gerçekleştirilmesinde sözkonusu katalog servisleri arasındaki birlikte işliğin nasıl sağlanacağıdır.

Bu çalışmada, bilgi kaynaklarının kataloglanmasına yönelik olarak geliştirilmiş olan çeşitli teknolojilerin basitleştirici ve birleştirici bir çerçeveden anlaşılmasını sağlayan, gerçekleştirmeci ve karar vericiler için yol gösterici olan bir dizi ölçüt belirlenmiştir. Konumsal ve konumsal olmayan bilgi toplumu tarafından geliştirilen katalog servisi teknolojileri, belirlenen ölçütlere göre değerlendirilmiştir. Ayrıca yukarıda belirtilen sorunlarla ilgili olarak elde edilen çözüm önerileri özetlenmiştir.

2. SERVİS KATALOGLAMA

2.1. Servis Katalogları ve Katalog Servisleri

Servis katalogları, SyM'nin kilit bileşeni durumundadır. Çünkü SyM'de servis sağlayıcılar Web servislerini katalog servisleri üzerinden kullanıma sunar, istemciler de aradıkları özellikteki Web servislerini bu sayede tespit ederler. Diğer bir anlatımla katalog servisi, sağlayıcıların Web servislerini katalog servisine kaydetmelerine ve istemcilerin istenen özellikleri sağlayan Web servislerini aramalarına olanak tanır. Burada söz konusu olan iki işlem, sırasıyla, SyM'nin meşhur "yayınla" (publish) ve "bul" (find) işlemleridir. Ancak Colan (2004) tarafından da işaret edildiği üzere, mevcut Web servisleri uygulamalarında katalog bileşeninin genellikle eksik olduğu görülmektedir. (W3C, 2004) gibi W3C'nin ilgili dokümanlarında bile konu yeterince detaylandırılmamıştır. Diğer bir anlatımla, istemcinin aradığı özellikteki servisleri bir şekilde "bulmuş olduğu" kabul edilmektedir. Oysa SyM'nin potansiyelleri ve sorunları bakımından, gerek teorik ve gerekse pratik olarak değerlendirilebilmesinde, onun en temel bileşeni olan katalog boyutundaki değerlendirmeler son derece kritik bir rol oynayacaktır.

SyM ve onun halen en geçerli gerçekleştirim şekli olan Web servisleri zaten, bütün işlerin Web servisleri ile yürütüldüğü işleyişler için öngörülmüştür; Çeşitli sunucuların Web servislerini sunduğu ve istemcilerin de bu servisleri kullanarak uygulamalarını gerçekleştirdiği bir işleyiş. Bu açıdan, Ryman'ın (2000) hava alanında "check-in" yaptırırken biletinin, bir şekilde birden çok müşteriye satılmış olduğunu fark eden yolcunun, biletinin "doğrulanmış" olup olmamasına ve kendi seyahat planına göre, yeni bilet arama ya da doğrulanmış biletini açık artırmada satma işlemlerini, Web servisleri aracılığı ile gerçekleştirebilme örneği, çok iddialı gibi görünse de Web servisleri ile hedeflenen ortamdır. Kullanıcıların Web servisleri ile böyle "iddialı" uygulamalar gerçekleştirebilmeleri için, çok yaygın bir servisler ağı ile desteklenmiş olmaları gerekir. Böyle bir ortam buradan itibaren "Yaygın Web Servisleri Ortamı" (YWSO) olarak anılacaktır. Anılan örnekte en azından havayolları, otel ve

oto kiralama sektörlerinin Web servislerine ihtiyaç olacaktır. Benzer şekilde UKVA'da da çok sayıda servis ve bunların sağlayıcıları söz konusu olacaktır. Bu durumda bir istemci gereksinimlerini karşılayacak özelliklerdeki bir ya da daha çok servisi nasıl bulacaktır? Bunun ne gibi boyutları vardır? Bu bağlamda katalogların önemi nedir? gibi sorular bu çalışmada ele alınacak konunun özünü teşkil etmektedir.

Gerçekten de Web servisleri vizyonunda kritik nokta, istemcilerin aradıkları özellikleri sağlayan servisleri bulmaları ile ilgilidir. Aranılan servis ya da servisler bulunduktan sonra geriye yalnızca servisin icrası ya da birden çok servis olması durumunda, servislerin belirli bir düzenleme ile icrası kalmış olacaktır. O halde servis bulmanın ne içerdiğine bakmak gerekir. Servis bulmanın her şeyden önce bir "sözdizimsel" (syntactic) bir de "anlamsal" (semantic) boyutu vardır. Daha sonra, "bir servisi ne gibi özellikler tanımlar?" sorusu ile alakalıdır. Servisin örneğin hangi tip veriler için tanımlı olduğu bilgisi, servis tanımlayıcı bir özellik olarak kullanılmalı mıdır? Bir diğer boyut YWSO da, servislerin nasıl kataloglanacağıdır. Servisler merkezi bir katalog da mı depolanacaktır yoksa birden çok katalog mu olmalıdır? Birden çok ve farklı katalog kullanılması durumunda birlikte işlerlik sağlanabilecek midir? Bütün bu sorunlara yönelik ne gibi çözümler ve araçlar mevcuttur ve bunlar bir çerçeve de değerlendirilebilir mi?

Web servisleri alanında Dünya genelindeki müthiş aktiviteye rağmen, ne konumsal Web servisleri ne de çok daha geniş bir kitle oluşturan konumsal olmayan Web servisleri alanında, YWSO lerin nasıl gerçekleştirileceği konusunda üzerinde görüş birliği sağlanmış bir gerçekleştirim modeli mevcuttur. Gerçekleştirim modeli bir yana, anılan sorunların bütüncül bir yaklaşımla ele alınmasını sağlayacak bir çatı bile mevcut değildir. Öyle ki Web servisleri alanında belirtim ve standart geliştirme ile ilgili çeşitli uluslararası kuruluşun (W3C, OASIS, WSMO, OGC) her birinin kendi çalışmaları vardır. Bunların hangilerinin yaygın kabul göreceği, farklı standartlar arasında nasıl uyum sağlanacağı henüz belirsizdir. Akademik çalışmaların çoğu ya çok genel ya da çok özeldir. Konumsal ve konumsal olmayan Web servisleri alanında faaliyet gösteren yazılım firmalarının kendi Web servisleri geliştirme ve yayma ortamlarının birlikte işlerliği henüz test edilmemiştir. Diğer yandan piyasada çeşitli gerçekleştirimler mevcuttur. Bu durumda Web servisleri yönelimli KVA gerçekleştiricileri nasıl bir yol izlemelidir?

Anılan sorunların basitleştirici ve bütüncül bir yaklaşımla ele alınmasına yardımcı olacak bir çatıya yönelik olarak kataloglama bağlamında bir takım ölçütler belirlenebilir mi? noktasından hareketle bu çalışmada bir dizi ölçüt belirlenmiştir. Konumsal veri ve servis boyutlarını içeren böyle bir çalışma mevcut değildir. Bununla birlikte, konumsal olmayan veri ve servislere yönelik bazı çalışmalar mevcuttur. Paolucci vd. (2004) de bu eksikliğe dikkat çekmektedir. Az sayıdaki çalışmalardan biri olan (W3C, 2004), Web servisleri mimarisine yönelik olarak bir dizi gereksinim belirlemiştir. Bunların bir kısmı katalog gereksinimleri ile aynıdır. Dustdar ve Treiber (2005), bu noktadan hareketle (W3C, 2004) ölçütlerini genişletmişler bazılarını da çıkararak 12 ölçüt önermişlerdir. Bu ölçütler oldukça tutarlı bir çerçeve çizmekle birlikte, bu çalışmanın amaç ve kapsamını aşan "güvenlik" ve "sağlamlık (fault tolerance)" gibi ölçütler içermekte, bu çalışma için çok gerekli olan federasyon gibi konularla ilgili ölçütler bakımından da eksik durmaktadır. Çünkü söz konusu çalışmada tek katalog değerlendirmesi üzerinde durulmuştur. Ayrıca, Dustdar ve Treiber (2005), "bakış noktası" bazlı bir değerlendirme uygulamış ve "insan bakış açısı" ve "Web servisi bakış açısı" itibarı ile bir değerlendirme yapmıştır. Ardından, bu bakış açılarının da "mimari" ve "veri modeli" gibi iki farklı boyuta işaret ettiklerini belirtmiştir. Ancak bu bakış açılarının seçimi gerekçelendirilmemiştir. Najmi'nin (2005) LDAP, UDDI, ebRIM ve ISO 11179 standartlarını karşılaştıran "yetenekler matrisi" çalışması, bu çalışmanın amacına göre çok detaylı olması bir yana, tümüyle birbirini dışlayan bir sınıflandırma içermektedir. Bunun bir nedeni, çalışmanın henüz devam ediyor olması olabilir. Ayrıca Sivashanmugam vd. (2004) gibi, yeterince detaylı olmayan, örneğin "federasyon desteği" gibi, yalnızca belirli noktalardan ebRIM ve UDDI kataloglarını karşılaştıran bazı çalışmalar mevcuttur. Konumsal olmayan çalışma alanlarından olan bu çalışmaların hiçbirinde doğal olarak konumsal veri boyutu yoktur.

Bu çalışmada, bilgi kaynaklarının kataloglanmasına yönelik olarak geliştirilmiş olan çeşitli teknoloji ve tekniklerin basitleştirici ve birleştirici bir çerçeveden anlaşılmasını sağlayacak, gerçekleştirimeci ve karar vericiler için yol gösterici olacak bir dizi ölçüt belirlenmiştir. Bu ölçütler, bir ya da daha çok katalog kullanılması gibi iki ana sınıf altında düşünülmüştür. Aslında amaç başlı başına UKVA'nın gerçekleştirimi ise, tek katalog seçimi çok önemli görülmeyebilir. Çünkü o durumda birden çok katalog kullanılması gerekecektir. Dolayısıyla bu katalogların birlikte işlerliği çok daha önemli olacaktır. Ancak UKVA gibi çok iddialı projelere geçişin, genellikle gözlemlendiği üzere, çok da eş zamanlı bir şekilde sağlanamaması nedeni ile tek katalog gerçekleştirimleri de önem kazanmaktadır. Diğer yandan bu ölçütler, piyasadaki ticari katalog ürünleri için, genel bir "benchmark" niteliği taşımaktadır. Sonuç olarak, tek katalog sınıfında katalogun bilgi modeli, katalog arayüzleri, katalog uygulama profili ve anlamsal destek ölçütleri belirlenmiştir.

2.2. Kataloglama Ölçütleri

2.2.1. Bilgi Modeli

Bir katalogun “bilgi modeli” aslında onun veri modelidir. Ancak kataloglar bağlamında kullanılan terim genellikle “bilgi modeli” olduğu için burada da bu isimlendirmeye bağlı kalınmıştır. Bu belki kataloglarda tutulan verinin de “bilgi kaynağı” olarak adlandırılmış olmasından kaynaklanmıştır. Bilgi modeli kendi içerisinde de çeşitli ölçütler içerir. Bunlar bilgi kaynağı türü, anlatım gücü ve genişletilebilirliktir. Aşağıda bunların her biri açıklanacaktır.

2.2.1.1. Bilgi Kaynağı Türü

Yukarıda da belirtildiği üzere, YWSO lar henüz erişilebilmiş bir hedef değildir. Bu nedenle, bugün bütün “sağlayıcılar” Web servisi sağlayabilecek durumda değildir. Bazı sağlayıcılar yalnızca veri sağlayabilir durumdadır. Bu nedenle mevcut uygulamada kataloglarda yalnızca Web servisleri değil, aynı zamanda XML şema dokümanları, “yetenekler” dokümanı, biçimlendirme dili dokümanları, çoklu ortam dosyaları ve belirtim dokümanları gibi çeşitli türde sayısal veriler kataloglanmaktadır. Bu bakımdan mevcut uygulamada kataloglarda tutulan bilgi çoğunlukla Web servisi olarak değil de “elektronik kapsam” ya da daha genel ifadeyle “bilgi kaynağı” olarak anılmaktadır (OASIS, 2005a; OGC, 2006). Burada en önemli sorulardan biri, “veriyi sunan servis varken o veriyi ayrıca sunmaya gerek var mıdır?” sorusudur. Bölümün başlığı da bu soru ile ilintilidir; yalnızca servisleri kataloglayan servis katalogları mı yoksa hem servis hem de veri kataloglayan katalog servisleri mi? Mevcut uygulama ve bu ölçüt açısından arzu edilen katalog servisleridir. Ancak, W3C Web servisleri mimarisine dayalı bir YWSO ya da UKVA ortamında söz konusu sorunun cevabı açık bir “hayır” dır.

2.2.1.2. Anlatım Gücü

Bilgi kaynakları, onları özet olarak tanımlayan ve “meta veri” olarak anılan özellikleri ile bir kataloga kaydedilir. Belirli özelliklerde bilgi kaynağı arayanlar da bu özellikler yardımı ile arama gerçekleştirirler. Bir Web servisini tanımlayabilecek sayısız meta verilerden bir kaç örneğin Web servisini sunan servis sağlayıcının hangi ticari sektöre ait olduğu, Web servisinin hangi tarihler arasında kullanıma açık olacağı, servisin hangi sınıflandırma sisteminde ne tür bir servis olduğu, servis sağlayıcının servis sunma kalite göstergesi olabilir. Bir veri grubu ise örneğin konumsal referans sistemi, veri kalitesi gibi özellikleri ile karakterize edilebilir. Hatta meta veriler arasındaki örneğin “bir Web servisi hangi tip bir Web servisinden sonra icra edilebilir?” ya da “bir servis hangi tip verileri kullanabilir?” gibi ilişkilerin temsil edilmesi gerekebilir.

Söz konusu tanımlayıcı özellikler ve ilişkiler aslında belirli “kavramsallaştırmalara” işaret eder. Bu kavramsallaştırmaların neler olması gerektiğini uygulamalar ya da kullanıcı kitlesi belirleyecektir. Bu kavramsallaştırmaların ne oranda temsil edilebildiği ise, kataloglamada kullanılan meta veri modelinin ya da katalogun “bilgi modeli” nin “anlatım gücü”ne bağlı olacaktır. Geleneksel veri modellemede kavramsallaştırmalar belirli temel kavramlarla ifade edilir. Bunlar genelleştirme, ilişki kurma (association), oluşum (composition) ve bir araya getirme (aggregation) dir. UML (Unified Modelling Language) (OMG, 2005) dili bunlara ilave olarak, örneğin “bağımlılık” gibi daha pek çok kavram içermektedir. İlişki kurma iki ya da çok varlık arasında bir ilişki tanımlar. Aslında oluşum ve bir araya getirme de birer ilişkidir. Her ikisi de varlıklar arasında “parça-bütün” ilişkisi tanımlar. Aradaki fark, oluşumda bütün durumundaki nesnenin silinmesi, parçalar anlamsız durumda kalacağı için, parçaların silinmesini de gerektirir. Bir araya getirme de böyle bir durum yoktur. Örneğin gölet ve akarsu nesnelerinin bir araya getirilmesi ile oluşturulan bir “drenaj-ağı” nesnesinin silinmesi parça ya da bileşen durumundaki gölet ve akarsuların silinmesini gerektirmez. Genelleştirme genellikle özelleştirme ile birlikte anılır. Örneğin ISO 19115 meta veri standardında “*MD_identification*” sınıfı “*MD_dataIdentification*” ve “*SV_ServiceIdentification*” sınıflarının her ikisinin de üst sınıfı yani daha genel halidir. Genelleştirmenin ters yönü ise özelleştirme ifade eder.

“Anlatım gücü”, veri modelleme literatüründe sıkça anılan bir konu olmakla birlikte, gerek tanımı ve gerekse anlatım gücünün neden yüksek olması gerektiği çok somut açıklanamamaktadır. Batini vd. (1992), bir şemanın anlatım gücünü, şemanın “Varlık-İlişki” (V-İ) modeli kavramları yardımıyla doğrudan, ilave bir açıklamaya gerek kalmadan anlaşılabilmesine bağlı olarak tanımlamaktadır. Saltor vd. (1991), bir veri modelinin anlatım gücünü, modelin bir kavramsallaştırmayı doğrudan temsil edebilme derecesi olarak tanımlamakta ve konunun “yapısal” ve “davranışsal” iki farklı boyutuna işaret etmektedir. Tabii, bu yararlı bir sınıflandırmadır ancak, konunun meta veri modelleme açısından öneminin irdelenebilmesi için, biraz daha somutlaştırılması gerekir. Çünkü “doğrudan temsil etme” hala oldukça soyut durmaktadır.

Aslında konunun iki boyutu vardır; birincisi kavramsal veri modeli ikincisi ise gerçekleştirim veri modeli boyutudur. Kavramsal veri modeli boyutunda modelin içerdiği kavramlar ne kadar fazla ise anlatım gücü de o kadar yüksek olacaktır. Örneğin V-İ modelinin anlatım gücü, ilişki modeline göre daha yüksektir. Bu durumda V-İ

modelindeki bir şema eğer ilişkisel olarak eşdeğerine dönüştürülerek gerçekleştirilecekse bazı kavramlar kaybolabilir. Örnek olarak V-İ modelindeki “zayıf varlık tipi” alınabilir. İlişkisel şemaya geçişte bu kavram kaybolacağı için, kavramın ifade ettiği anlamın yakalanması ancak uygulama programları ile mümkün olacaktır ki, bu tip işlevler bir yandan uygulama programlarının yükünü artırırken, bir yandan da veri tabanının yönetimini zorlaştıracaktır. Oysa aynı kavram V-İ modeli gibi varlık tabanlı bir yapıda örneğin Nesne Yönelimli bir veri tabanı ve Nesne Yönelimli Programlama (NYP) ile gerçekleştirilirse, aynı anlamın gerektirdiği davranış, nesnelere vasıtasıyla doğrudan gerçekleştirilebilecektir. Çünkü NYP, yeniden-kullanılabilirliğin çok üst düzeyde olması nedeni ile de davranış modellemede kolaylık sağlamaktadır (Meyer, 1988). Bununla birlikte, bu ölçütle kastedilen kavramsal düzeydeki anlatım gücüdür.

Katalog servisleri bilgi modeli olarak ya kendi modellerini ya da bir meta veri modelini kullanabilirler. ISO 19115 (ISO, 2003) ve FGDC kapsam standardı (FGDC, 1998) verilere yönelik, ISO 19119 (ISO, 2001) ise Web servislerine yönelik meta veri standartlarına örnektir. Bu ölçüt açısından arzu edilen, katalogun anlatım gücü yüksek olan bir bilgi modeline sahip olmasıdır.

2.2.1.3. Genişletilebilirlik

Genişletilebilirlik, bilgi modeline yeni kavramsallaştırmaların eklenebilmesindeki kolaylık olarak tanımlanabilir. Yeni kavramsallaştırmalar, bilgi modeline yeni sınıflar ya da ilişkiler eklenmesini gerektirebilir. Geleneksel veri modellemede bugün gelinen nokta itibarı ile yaygın kabul edilen bir gerçek, bütün ihtiyaçları karşılayacak genel bir model tasarlanmasının imkansız olduğudur (Wiederhold, 1992). Bu bakımdan, aynen veri modellemede olduğu gibi, meta veri modellemede de meta veri modelinin anlatım gücünün yüksek olmasından daha önemlisi belki de, modelin ihtiyaçlara kolaylıkla uyarlanabilir ya da “genişletilebilir” olmasıdır. Bu açıdan varlık bazında tanımlama ve ilişki kurma içeren “varlık tabanlı” modeller avantaj sağlamaktadır. Gerek genelleştirme ve gerekse oluşum hiyerarşileri ve ilişki kurma vasıtasıyla varlıklar arasında yeni ilişkiler kolaylıkla tanımlanabilir. Ancak ilişki kurma yoluyla kavramsallaştırmaların ve dolayısıyla, anlatım gücünün artırılması, bir yandan sorgulanabilir özellikleri artırırken, diğer yandan katalog gerçekleştirimini ve farklı katalog bilgi modellerinin kullanıldığı bir ortamda gerek söz dizimsel ve gerekse anlamsal birlikte işlevli zorlaştıracaktır.

2.2.2. Katalog Arayüzleri

Katalog arayüzleri, katalogun iki temel bileşeninden biri olarak düşünülebilir. Bu bileşenlerden biri katalog bilgi modeli, diğeri katalog arayüzleridir. Bilgi modeli, bilgi kaynaklarının katalogda nasıl tanımlanacaklarını belirler. Katalog arayüzleri ise bir bilgi kaynağını katalogda yayınlama, arama gibi işlevlerin yürütülmesine olanak tanırlar. Katalog arayüzleri açısından istenen ya da gereksinim duyulanan ne olduğu konusunda bilgimiz dahilinde bir çalışma yoktur. Ancak, SyM'nin temel “yayınla” (publish) ve “bul” (find) işlemlerinin gerçekleştirilebilmesi açısından bir katalog servisinin yayınlama ve bulma arayüzlerine sahip olması gerekmektedir. Bunlara ilave olarak katalog servisinin arayüz fonksiyonlarının fazlalığı artı bir puandır.

2.2.2.1. Bulma (Sorgu) Arayüzü

Sorgu arayüzü, istemcilerin katalogu sorgulamak için kullandıkları arayüzdür. Sorgu arayüzü açısından istenen ya da gereksinim duyulanan ne olduğu konusunda bilgimiz dahilinde bir çalışma yoktur. Genel olarak arzu edilen, gerektiğinde “parametrik” ve “anlık” (ad-hoc) sorgu yapılabilmesidir. UDDI gibi bazı kataloglar yalnızca parametrik sorgu desteği vermektedir. Bu da kullanıcının aradığı özelliklerin “parametre” olarak önceden tanımlanmamış olması durumunda sorgu yapılamaması nedeni ile kısıtlayıcı bir yoldur. Anlık sorgulamayı destekleyen kataloglar genellikle bir filtreleme dili ya da SQL ile sorgulamaya izin verirler (OGC, 2005c; OASIS, 2005a). Bu durumda da “sorgulama dilinin anlatım gücü” (Dustdar ve Treiber, 2005) önem kazanmaktadır. Bir başka ifadeyle sorgu dilinin anlatım gücünün yüksek olması istenir. Konumsal veri açısından, sorgulama dilinin OGC CSW filtreleme dilinde olduğu gibi konumsal operatörleri desteklemesi gerekir.

2.2.2.2. Yayınlama Arayüzü

SyM de “yayınlama” işlemi ile kastedilen, sağlayıcıların, sahip oldukları bilgi kaynaklarını tanımlayan meta verileri, bir katalog servisine koyarak, servis ya da veri kaynaklarını başkalarının kullanımına açmalarıdır. Sağlayıcılar bu işlevi, katalogun yayınlama arayüzünü kullanarak yerine getirirler. Yayınlama arayüzü açısından arzu edilebilecek noktalardan biri, katalogun hem sağlayıcıların doğrudan yayınlama işlemi yapmasına, hem de katalogun tüm kullanıcılardan yayın toplaması yolu ile yayınlama yapılmasına olanak tanımasıdır.

2.2.2.3. Diğer Arayüzler

Katalog arayüzleri açısından istenen ya da gereksinim duyulanın ne olduğu konusunda bilginiz dahilinde bir çalışma yoktur. Arayüz fonksiyonlarının fazlalığı artı bir puandır. Örneğin ebXML katalog servisinin sunduğu “olay bildirme” (event-notification) fonksiyonu katalog içeriğindeki değişiklikleri abone olan kullanıcılara bildirmektedir. Bu sayede, örneğin katalogda yeni yayınlanmaya başlanan ya da katalogda yayınlanması durdurulan servislerin bu durumları kullanıcılara bildirilebilir.

2.2.3. Katalog Uygulama Profili

OGC (2005a), CSW bağlamında, bir katalog uygulama profilini, katalogun belirli bir protokole göre olan gerçekleştirimi olarak tanımlamaktadır. CSW belirtimi, katalog servisi arayüzlerinin Z39.50, CORBA/IIOP ve HTTP protokollerindeki gerçekleştirimlerini tanımlamaktadır. Bir CSW katalog servisinin gerçekleştirebileceği arayüzler, katalog servisinin kullandığı uygulama protokolüne bağlı olarak değişir. Örneğin, HTTP protokolü “durumsuz” (stateless) bir protokol olduğu için, bu protokolü kullanan bir CSW katalog servisi “Oturum” (Session) arayüzünü gerçekleştiremez. Temel belirtim, uygulama profili ve katalog servisi arasındaki ilişkiyi, OGC (2005a)’da, “...bir katalog servisi belirtimine uyan birden çok uygulama profili ve bir uygulama profiline uyan birden çok katalog servisi olabileceği...” şeklinde özetlenmiştir. Çünkü bir uygulama profili oluşturmada, bütün durumlarda olmasa bile çoğunlukla söz konusu olan ana modelin daraltılması ya da özelleştirilmesidir.

Uygulama profili ile kastedilen, katalogun belirli bir gerçekleştirim topluluğu (domain) için özelleştirilmesidir. Bu özelleştirme bazen genişletme de içerebilir. Bu durumda birden çok katalogun aynı sorgu ile sorgulanabilmesi için sorgunun ortak sorgulanabilir özellikler üzerinden yapılması gerekir. Uygulama profilleri bazındaki farklılıklar ise ancak farklı protokollere sahip olan sağlayıcılardaki dönüştürücülerle sağlanabilir. Bu durumda özetle söylenecek şey, farklı uygulama profili sayısı arttıkça birlikte işlerliği sağlamanın zorlaşacak olduğudur.

2.2.4. Anlamsal Destek

Bu ölçüt katalog bilgi modelinin anlamsal olarak genişletilebilmesine yöneliktir. Anlamsal destek aslında genişletilebilirlik ölçütü altında da düşünülebilirdiyse de, katalogun “Anlamsal Web” (Berners-Lee vd., 2001) açısından kullanılabilirliğinin değerlendirilebilmesi açısından ayrı bir ölçüt olarak düşünülmüştür. Bu ölçüt açısından arzu edilen, katalog servisinin, bilgi kaynağı tanımlamalarını anlamsal olarak zenginleştirmeye olanak sağlayan bir bilgi modeline sahip olmasıdır.

3. KATALOG SERVİSLERİNİN DEĞERLENDİRİLMESİ

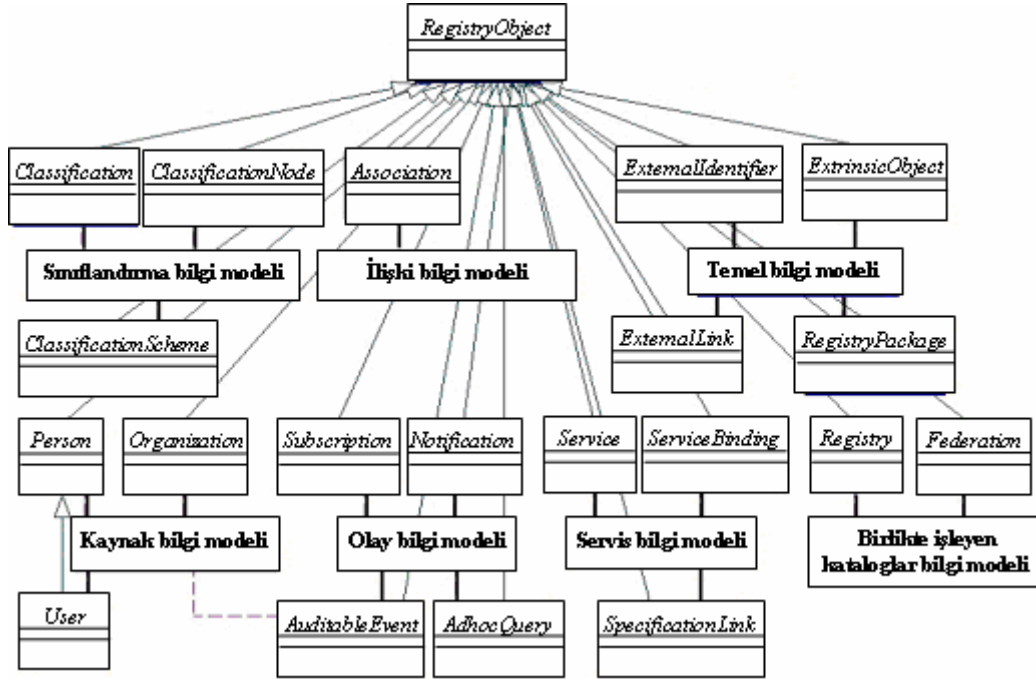
Çalışmanın bu bölümünde, OASIS (Organization for the Advancement of Structured Information Standards) ortaklığı tarafından geliştirilen ve standart olarak kabul edilen UDDI (Universal Description Discovery and Integration) ve ebXML Katalog Servisi (ebXML Registry) ile Open Geospatial Consortium (OGC) tarafından geliştirilen CSW katalog servisi ve bu servisin ISO 19115/19119 ve ebRIM profilleri yukarıda açıklanan ölçütlere göre değerlendirilecektir. Değerlendirme sonuçlarının, bu bildiri kapsamında verilemeyecek kadar detaylı ve uzun olması nedeniyle bu bölümde her bir ölçüt ile ilgili kısa ve öz bilgiler verilecektir. Katalog servislerinin ayrıntılı değerlendirmesi (Akıncı, 2006) da bulunabilir.

3.1. ebXML Katalog Servisi

ebXML girişimi, elektronik-iş’i (e-iş) desteklemek için XML tabanlı bir altyapı geliştirmek amacıyla, UN/CEFACT (United Nations Centre for Trade Facilitation and Electronic Business) ve OASIS ortaklığı tarafından Kasım 1999 da başlatılan ve Mayıs 2001 de tamamlanan bir projedir (Manes, 2002). Bu ortaklık tarafından geliştirilen e-iş altyapısı, bir kısmı OASIS, bir kısmı da UN/CEFACT tarafından yönetilen belirtilerle tanımlanmıştır. OASIS, ebXML Registry Teknik Komisyonu’nu Mayıs 2001 de kurmuş ve ebXML Katalog Servisinin (ebXML KS) ikinci sürümü, teknik komisyon tarafından Nisan 2002 de standart olarak onaylanmıştır. ebXML KS’nin, şu an yürürlükte olan son sürümü komisyon tarafından Mayıs 2005 de onaylanmıştır. ebXML KS, birbirini tamamlayan iki ayrı standart tarafından tanımlanır. İlk standart, “ebXML Registry Services and Protocols” (ebRS) (OASIS, 2005a), bir ebXML KS’nin desteklemesi gereken arayüzleri, ikinci standart ise, “ebXML Registry Information Model” (ebRIM) (OASIS, 2005b), ebXML KS’nin bilgi modelini tanımlar.

Bilgi modeli: ebXML KS’nin bilgi modeli, nesne yönelimli bir tasarıma sahiptir. Bilgi modeli, her türlü bilgi kaynağını tanımlamaya olanak sağlayan otuzdan fazla meta veri sınıfı içerir. Bilgi modelinde yer alan meta veri sınıflarının birçoğu, *RegistryObject* sınıfının bir alt sınıfıdır (Şekil 2). *RegistryObject* sınıfı, diğer sınıfların bilgi kaynaklarını tanımlamak için kullandıkları ortak meta veri özelliklerini sağlayan bir üst sınıftır. Bir katalog servisinin bilgi modeli, bilgi kaynağı türü, anlatım gücü ve genişletilebilirlik ölçütlerine göre değerlendirildiği için,

bu bölümde sadece ebXML KS bilgi modelinin söz konusu ölçütler açısından değerlendirilmesine olanak sağlayan meta veri sınıfları açıklanacaktır. Bilgi modelindeki diğer sınıflarla ilgili detaylı bilgiler, ebRIM standardında (OASIS, 2005b) bulunabilir.



Şekil 2. ebXML katalog servisi bilgi modeli (OASIS, 2005b).

Bilgi kaynağı türü: ebXML KS, her türlü bilgi kaynağını tanımlamaya olanak sağlayan bir bilgi modeline sahiptir. Bir ebXML KS, sicil (registry) ve depo (repository) olarak adlandırılan iki ana bileşenden oluşur. ebXML KS terminolojisinde, WSDL, BPEL gibi XML dosyaları, GIF, JPEG gibi görüntü dosyaları ve WAV, MPEG gibi çoklu ortam dosyalarını da içeren her türlü elektronik doküman, “depo elemanları” (repository items) olarak adlandırılır ve ebXML KS’nin depo bileşeninde yer alır. Depo elemanlarını tanımlamak için kullanılan meta veri sınıfları ise, “sicil nesnelere” (registry objects) olarak adlandırılır ve ebXML KS’nin sicil bileşeninde yer alırlar (OASIS, 2005b).

Anlatım gücü: ebXML KS, Nesne-Yönelimli bir bilgi modeline sahiptir ve veri modellemede kullanılan genelleştirme, ilişki kurma, oluşum ve bir araya getirme gibi kavramları içermektedir. Bu nedenle ebXML KS bilgi modelinin anlatım gücü yüksektir. ebXML KS bilgi modeli ayrıca, sicil nesnelere arasında ilişki kurmaya olanak sağlayan “Association” sınıfına sahiptir. Association sınıfının “sourceObject” ve “targetObject” öznitelikleri, aralarında ilişki kurulacak olan sicil nesnelere belirtir. Sicil nesnelere arasındaki ilişkinin tipi, Association sınıfının “associationType” özneliği ile belirlenir. ebRIM standardı, bir ebXML KS nin kullanacağı ilişki tiplerini önceden tanımlanmıştır. Örneğin, bir Web servisi ile servisi sunan organizasyon arasındaki ilişki “SubscriberOf” ilişki tipi ile tanımlanmaktadır. Bilgi modeli, kullanıcıların yeni ilişki tipleri tanımlamasına olanak tanımaktadır.

Genişletilebilirlik: ebXML KS, genişletilebilir bir bilgi modeline sahiptir. Slot sınıfı, bilgi modelinin genişletilebilirliğine olanak sağlayan en önemli sınıftır. Bir kullanıcı, Slot sınıfını kullanarak bilgi modelindeki herhangi bir sicil nesnesine ilave öznitelikler ekleyebilir. Bilgi modelinin genişletilebilirliğine olanak sağlayan bir diğer sınıf, Association sınıfıdır. Kullanıcılar, ebRIM standardı tarafından önceden tanımlanan ilişki tiplerinin, sicil nesnelere arasındaki ilişkileri tanımlamak için yetersiz kaldığı durumlarda, yeni ilişki tipleri tanımlayarak bilgi modelini genişletebilirler.

Katalog Arayüzleri: ebXML KS, kullanıcılarına oldukça zengin bir arayüz desteği sunmaktadır. Ancak bu bölümde, ebXML KS’nin bulma ve yayınlama arayüzüne ilave olarak “yaşam süreci yönetim” (lifecycle management) arayüzü ve “olay bildirme” (event notification) arayüzü açıklanacaktır. ebXML KS’nin bu çalışma kapsamında ele alınamayan, örneğin “güvenlik arayüzü” gibi diğer arayüzleri ile ilgili bilgiler, ebRS standardında bulunabilir.

Bulma arayüzü: İstemciler, bir ebXML KS tarafından yayımlanan bilgi kaynaklarını bulmak için ebXML KS’nin “sorgu yönetim” (query management) arayüzünü kullanırlar. Sorgu yönetim arayüzü, istemcilerin iki türlü sorgu yapmalarına olanak sağlar. İlk sorgu yöntemi “anlık sorgu” (ad hoc query), ikincisi ise “saklanmış sorgu” (stored query) olarak adlandırılır. Anlık sorguda istemci bir sorgu mesajı yazar ve bir ebXML KS’ye gönderir. Sorgu mesajı, bilgi kaynaklarını tanımlamak için kullanılan meta veri sınıflarının özniteliklerinden oluşan ve SQL ya da

ebRIM Filter sözdizimine göre yazılan bir sorgu ifadesi içerir. Bir ebXML KS kullanıcısı, SQL ve ebRIM Filter sözdizimini veya ebXML KS'ye kayıtlı taksonomileri ve bu taksonomilerin kategorilerini bilmeyebilir. Bu nedenle, istemcilerin bir ebXML KS'yi kolay bir şekilde sorgulayabilmesi için, istemciler tarafından sıkça kullanılacağı düşünülen sorgular veya birden fazla meta veri sınıfının özneliklerini içeren ve istemciler tarafından yazılması zor olan karmaşık sorgular, ebXML KS yöneticisi tarafından önceden yazılır ve bilgi modelinin *AdhocQuery* sınıfı kullanılarak ebXML KS'ye kayıt edilir. Bu şekilde ebXML KS'ye kaydedilen sorgulara "saklanmış sorgu" adı verilir. ebXML KS, istemcilerin içerik tabanlı (content-based) sorgular yapmasına da olanak sağlamaktadır. Ancak istemcilerin içerik tabanlı sorgu yapabilmesi için, içeriğin bir "içerik yönetim servisi" (content management service) tarafından önceden kataloglanmış olması gerekmektedir. İçerik kataloglama servisi, istemcilerin ebXML KS'ye yayınladıkları dokümanların (içeriğin), ebXML KS tarafından otomatik olarak kataloglanması sağlar.

Yayınlama arayüzü: Bir sağlayıcı, sahip olduğu bilgi kaynaklarını bir ebXML KS'ye yayınlamak için, ebXML KS'nin yaşam süreci yönetim arayüzünü kullanır. Yaşam süreci yönetim arayüzünün "SubmitObject" metodu, sicil nesnelere ve depo elemanlarının bir ebXML KS'ye kayıt edilmesini sağlar.

Diğer arayüzler: ebXML KS'nin kullanıcılara sunduğu önemli arayüzlerden birisi de "olay bildirme" arayüzüdür. Olay bildirme arayüzü, ebXML KS de belirli bir olayın gerçekleşmesi durumunda, bu olay ile ilgilenen kullanıcıların haberdar edilmesine olanak sağlar. Örneğin istemci, ebXML KS'ye yeni bir WFS servisinin yayınlanması durumunda veya kullandığı bir Web servisinin güncellenmesi durumunda, ebXML KS'nin kendisini uyarmasını isteyebilir. ebXML KS, bir olayın gerçekleştiğini kullanıcıya genellikle e-mail göndererek bildirir.

Katalog uygulama profili: ebXML katalog servisleri, özelleştirilebilir bir bilgi modeline sahiptir. ebRIM bilgi modelinin özelleştirilmesini gerektiren iki durum söz konusudur. Birincisi, ebXML katalog servislerinin farklı ilgi alanları (domain) tarafından kullanılmak istenilmesi durumunda bilgi modeli ilgi alanının ihtiyaçlarını karşılamak için özelleştirilebilmektedir. ebRIM bilgi modelinin bir ilgi alanına özel gerçekleştirimi, genellikle bir "uygulama profili" tarafından tanımlanmaktadır. Örneğin, ebXML katalog servislerinin, sağlık hizmetleri alanında kullanılabilmesi için HL7 (Healthcare Level 7) profili geliştirilmiştir. Bilgi modeli ikinci olarak, farklı ilgi alanları tarafından kullanılan katalog servislerine bilgi modeli desteği sağlamak için özelleştirilebilmektedir. Örneğin OGC, CSW katalog servislerinin bilgi modelini tanımlamak için CSW-ebRIM profilini geliştirmiştir (OGC, 2006).

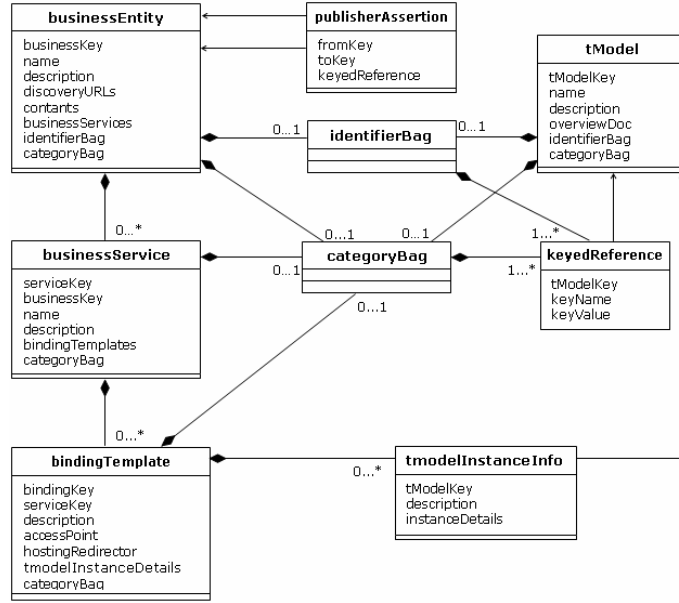
Anlamsal destek: ebXML KS, Web servisi tanımlamalarını anlamsal olarak zenginleştirmeye olanak sağlayan bir bilgi modeline sahiptir. Bu alandaki başlıca çalışmalar (Doğaç, 2003; Doğaç vd., 2003; Doğaç vd., 2004; Doğaç vd., 2005) olarak sayılabilir. Bu çalışmalarda genel olarak iki yolla anlamsal zenginleştirme uygulanmıştır. Doğaç (2003) ve Doğaç vd., (2003) te uygulanan ilk yöntemde, Web servisleri belirli "alan ontolojilerine" (domain ontologies) dayalı olarak DAML-S dilinde tanımlanmış ve bu tanımlamalar, ebRIM bilgi modelinin *ClassificationScheme* ve *ClassificationNode* sınıflarına resmedilmiştir. Doğaç vd. (2004) ve Doğaç vd. (2005) gibi daha sonraki çalışmalarda uygulanan ikinci yöntemde ise, Web servisleri arasında kullanılan ontoloji gereği temsil edilmesi istenen ilişkilerin, *Association* nesnelere vasıtası ile temsil edilmesi yoluna gidilmiştir. Literatürde yukarıda anılan çalışmalara oranla daha yeni bir çalışma olarak tespit edilen Di vd. (2006)'nın çalışması, konumsal verilere yönelik olması ile Doğaç ve ekibinin yukarıda anılan çalışmalarından farklı, ancak anlamsal desteğin bir ebXML KS de nasıl ve ne dereceye kadar temsil edilebileceğinin değerlendirilmesi açısından benzerdir. Söz konusu çalışmalarla ilgili detaylı bilgiler (Akıncı, 2006) da bulunabilir.

3.2. UDDI

UDDI, Microsoft, IBM ve Ariba yazılım firmalarının kurduğu, "UDDI.org" olarak adlandırılan bağımsız bir ortaklık tarafından geliştirilmiştir. UDDI belirtiminin ilk sürümü, Eylül 2000 de bu ortaklık tarafından bilişim dünyasına duyuruldu (Cerami, 2002). Herkesin kullanımına açık ilk UDDI katalog servisleri, Mayıs 2001 de Microsoft ve IBM firmaları tarafından hizmete sunuldu. UDDI belirtiminin yenilenen ikinci sürümü, Haziran 2001 de ilan edildi. UDDI.org, Temmuz 2002 de OASIS birliğine dahil oldu (Newcomer, 2002) ve UDDI'ın ikinci sürümü Mayıs 2003, şu an yürürlükte olan son sürümü (Version 3.0.2) ise Şubat 2005 de OASIS tarafından standart olarak kabul edildi.

Bilgi modeli: UDDI, Web servislerini ve servis sağlayıcılarını tanımlamaya olanak sağlayan bir bilgi modeline sahiptir. UDDI bilgi modeli, dört temel sınıftan oluşur. Bunlar, "businessEntity", "businessService", "bindingTemplate" ve "tmodel" sınıflarıdır (Şekil 3). *BusinessEntity* sınıfı, servis sağlayıcılar ile ilgili meta verileri tanımlamak için kullanılır. *BusinessEntity* sınıfı, ebRIM bilgi modelinde, bilgi kaynağı sağlayıcılarını tanımlamak için kullanılan "Organization" sınıfına karşılık gelir ve servis sağlayıcıları ile ilgili iletişim ve sınıflandırma bilgilerini içerir. Bir servis sağlayıcısı tarafından sunulan Web servisleri, *businessService* sınıfı kullanılarak tanımlanır. *BindingTemplate* sınıfı ise, bir Web servisini çağırma için gerekli olan bilgileri içerir. UDDI bilgi modelinin en önemli ve anlaşılması bir o kadar da zor olan sınıfı, "technical model" veya kısaca "tmodel" sınıfıdır. UDDI belirtimi (OASIS, 2004), *tmodel* sınıfının ne işe yaradığını ve görevinin ne olduğunu tam olarak ortaya koymamıştır. Bu durum literatürde de dile getirilmektedir (Cerami, 2002; Yu, 2006). *Tmodel*, bir tip tanımlama mekanizmasıdır ve servis sağlayıcılarının ve Web servislerinin tiplerini tanımlamaya olanak sağlamaktadır. *Tmodel*,

ebRIM bilgi modelinde taksonomileri tanımlamak için kullanılan *ClassificationScheme* sınıfına eşdeğerdir. *tmodel* sınıfı ayrıca, bir taksonomiyi temsil etmeyen belirtileri tanımlamak için de kullanılmaktadır. Örneğin bir Web servisini tanımlayan WSDL dokümanı, UDDI da *tmodel* sınıfı kullanılarak tanımlanır. ebRIM bilgi modelindeki *ExternalLink* nesnesinin, bir ebXML KS'nin depo bileşeninde yer almayan elektronik dokümanları tanımlamak için kullanıldığını ve dokümanların adreslerini içerdiğini hatırlarsak, *tmodel* sınıfının bu kullanım şekli ile de *ExternalLink* nesnesi ile eşleştiğini görmüş oluruz.



Şekil 3. UDDI bilgi modeli (UDDI.org, 2000).

Bilgi kaynağı türü: UDDI, ebXML KS de olduğu gibi her türlü elektronik dokümanı tanımlamaya olanak sağlayan bir bilgi modeline sahip değildir. UDDI, sadece Web servisleri ve servis sağlayıcıları ile ilgili meta verileri içeren bir katalog servisedir. Bu nedenle UDDI'ı “servis katalogu” olarak adlandırmak daha doğru olur.

Anlatım gücü: UDDI, ebRIM de olduğu gibi nesne yönelimli bir bilgi modeline sahip değildir. Literatürde, UDDI bilgi modelinin hiyerarşik bir yapıya sahip olduğu, bilgi modelindeki meta veri sınıfları arasında sadece içerme (containment) ilişkisinin olduğu ve bilgi modelinin bir XML şema tarafından tanımlandığı belirtilmektedir (UDDI.org, 2000; OASIS, 2004). Her ne kadar literatürde UDDI bilgi modeli için “hiyerarşik bir yapıya sahiptir” denilse de, aslında UDDI bilgi modeli “varlık” (entity) tabanlıdır ve varlıklar arasında sadece oluşum (composition) ve ilişki (association) ilişkisi vardır. Bu nedenle UDDI bilgi modelinin anlatım gücü zayıftır.

Genişletilebilirlik: UDDI, genişletilebilir bir bilgi modeline sahiptir ve kullanıcıların bilgi modelinde genişletme yapmalarına olanak sağlamaktadır. ShaikhAli (2003), UDDI bilgi modelinde bazı genişletmeler yapmış ve UDDIe (Extended UDDI) olarak adlandırılan açık kaynak kodlu bir UDDI katalog servisi geliştirmiştir.

Katalog arayüzleri: UDDI katalogları, yayınlama ve sorgu arayüzü gibi, bir katalog servisinin kullanıcılarına sunması gereken temel arayüzlerin yanı sıra, seçmeli olan güvenlik (security) arayüzüne, sorumluluk ve sahiplik transfer (custody and ownership transfer) arayüzüne ve abonelik (subscription) arayüzüne sahip olabilirler. Güvenlik arayüzü ile sorumluluk ve sahiplik transfer arayüzü, bu çalışmanın kapsamı içerisinde yer almadığı için bu bölümde söz konusu arayüzlere değinilmeyecektir. Bu arayüzler ile ilgili ayrıntılı bilgi UDDI belirtiminde (OASIS, 2004) bulunabilir.

Bulma Arayüzü: Bir UDDI katalog servisini sorgulamak ve Web servisleri bulmak için UDDI'nın sorgu (inquiry) arayüzü kullanılır. Web servisleri UDDI da, isimlerine göre, sağlayıcılara göre, sunuldukları coğrafik yere göre veya belirli bir taksonomiye göre aranarak bulunur. Örneğin bir istemci, katalogda, sağlayıcısı OMÜ Jeodezi ve Fotogrametri Mühendisliği Bölümü olan, Samsun'dan sunulan, ismi “Web Feature Service” olan ve NAICS taksonomisinin, “Bilgi Servisleri” (Information Services) kategorisine göre sınıflandırılan Web servislerini arayabilir. Ancak aynı servis, eğer “Türkiye’den sunulan servisler” kategorisinde de kataloga kaydedilmemişse, istemci servisi bu kategori altında bulamayacaktır. Çünkü ebXML KS den farklı olarak UDDI da taksonomiler temsil edilmemektedir. Bu açıdan ebXML KS daha esnekler. UDDI bulma arayüzü bir filtreleme ya da SQL arayüzü içermez. Dolayısıyla ebXML KS de mümkün olan “anlı” sorgulamalar UDDI da yapılamaz. UDDI da içerik tabanlı sorgulama da mümkün değildir. Bunun nedeni, UDDI'nın bir depo (repository) bileşenine sahip olmaması ve servislerin WSDL tanımlarının UDDI da değil de doğrudan servis sağlayıcıda bulunmasıdır.

Yayınlama arayüzü: Bir servis sağlayıcısı, sahip olduğu Web servislerini bir UDDI katalog servisine yayınlamak veya yayınladığı Web servisleri ile ilgili bilgileri güncellemek ya da silmek için UDDI’ın “yayınlama” (publication) arayüzünü kullanır. Servis sağlayıcı, yayınlama arayüzünün “*save service*” ve “*save binding*” metodlarını kullanarak servis bilgilerini, “*save business*” metodunu kullanarak sağlayıcı bilgilerini yayınlar. Adı geçen metodlar, parametre olarak, bilgi modelindeki ilgili sınıfların özniteliklerini alırlar. Örneğin *save_business* metodunun parametreleri, *businessEntity* sınıfının özniteliklerinden oluşur.

Diğer arayüzler: UDDI katalog servisleri, abonelik arayüzüne sahip olabilirler. Abonelik arayüzü, istemcilere bir UDDI katalogundaki değişiklikleri takip etme olanağı sağlar. Bir istemci, abonelik arayüzünün “*save_subscription*” metodunu kullanarak izlemek istediği değişikliklere abone olur. Örneğin istemci, UDDI da arama yaparak bulduğu ve kullandığı bir Web servisinin güncellenmesi durumunda veya belirli bir sağlayıcının kataloga yeni servisler yayınlanması durumunda, katalogun söz konusu değişiklikleri kendisine bildirmesini isteyebilir. İstemci, içerik değişikliklerinden iki şekilde haberdar olur. İstemci, ya abonelik arayüzünün “*get_subscriptionResults*” metodunu kullanarak değişiklik bilgilerini katalogdan kendisi alır ya da “*notify_subscriptionListener*” metodunu kullanarak değişikliğin katalog tarafından kendisine otomatik olarak bildirilmesini sağlar.

Katalog Uygulama Profili: UDDI, ebXML KS ve OGC CSW gibi katalog servislerine göre daha basit bir bilgi modeline sahiptir. Bir bilgi modelinin veya bilgi modeli olarak gerçekleştirilecek olan örneğin ISO 19115 gibi bir meta veri standardının, bir ilgi alanına özel gerçekleştirimi, genellikle bilgi modelinin veya meta veri standardının bir alt seti olan bir uygulama profili tarafından tanımlanır. UDDI bilgi modeli, diğer katalog bilgi modellerine göre oldukça az sayıda meta veri sınıfı içermektedir. Bir UDDI katalog servisi sağlayıcısının, UDDI bilgi modelinin tamamını gerçekleştirmesi, sağlayıcıya fazla bir yük getirmeyecektir. Bu nedenle UDDI’ın bir ilgi alanı için özelleştirilmesine veya bir UDDI profilinin tanımlanmasına gerek yoktur. Literatürde ve Web de yapılan araştırmalarda da, bir UDDI profiline veya UDDI’ın bir ilgi alanına özel gerçekleştirimine rastlanmamıştır.

Anlamsal destek: UDDI, Web servisi tanımlamalarını anlamsal olarak zenginleştirmeye olanak sağlayan bir bilgi modeline sahiptir. UDDI kataloglarına anlamsal destek kazandırmak için önemli akademik çalışmalar yapılmıştır (Paolucci vd., 2002a; Paolucci vd., 2002b; Paolucci vd., 2004; Doğaç vd., 2002a; Doğaç vd., 2002b). Paolucci vd. (2002a), Web servislerinin UDDI da sadece isimlerine göre değil de yeteneklerine göre de bulunmasına olanak sağlayan anlamsal bir servis bulma mekanizması geliştirmiştir. Paolucci vd. (2002a), DAML-S dilini kullanarak anlamsal olarak tanımlanan servis tanımlarını UDDI servis tanımlarına dönüştürmüştür. DAML-S servis tanımlarında yer alan ve UDDI servis tanımlarına doğrudan dönüştürülemeyen DAML-S servis parametreleri için UDDI da yeni bir *tmodel* oluşturulmuştur. Örneğin bir DAML-S servis tanımında yer alan sağlayıcı bilgileri, UDDI da *businessEntity* sınıfının özniteliklerine dönüştürülürken, servisin girdi ve çıktı parametrelerini tanımlamak için UDDI da “*input_tmodel*” ve “*output_tmodel*” olarak adlandırılan iki yeni *tmodel* yaratılmıştır. Oluşturulan bu yeni *tmodel* tanımlamaları Web servislerini sınıflandırmak için kullanılmıştır. Paolucci vd. (2002a) ayrıca, servis istekleri ile servis yetenekleri arasındaki anlamsal eşleşme düzeylerini hesaplamak için bir eşleştirme algoritması tanımlamış ve bu algoritmayı gerçekleştiren bir eşleştirici (DAML-S/UDDI Matchmaker) geliştirmiştir. Eşleştirici, UDDI kataloglarını, yetenek tabanlı eşleştirmeyi gerçekleştiren ilave bir anlamsal katmanla genişletmek için kullanılmaktadır.

3.3. OGC CSW Katalog Servisi

OGC, katalog servisleri ile ilgili çalışmalarına 1998 yılında başlamış ve Ağustos 1999 da katalog servis belirtiminin ilk sürümünü (OGC, 1999) yayınlamıştır. OGC’nin “Katalog Revizyon Çalışma Grubu”, kullanıcı gereksinimlerini ve bilgi teknolojilerindeki gelişmeleri dikkate alarak değişik zamanlarda katalog servisinde çeşitli düzenlemeler yapmış ve belirtimin yeni sürümlerini yayınlamıştır. Yürürlükte olan son sürüm (OGC, 2005b), Mayıs 2005 de yayınlanmıştır.

Bilgi modeli: CSW belirtimi, OGC uyumlu bir katalog servisinin desteklemesi gereken arayüzleri ve bu arayüzlerin farklı uygulama protokollerindeki gerçekleştirim esaslarını tanımlayan temel belirtimdir. CSW belirtiminde, CSW katalog servisinin bilgi modeli tanımlanmamıştır. CSW katalog servisinin bilgi modeli, OGC tarafından geliştirilen bir “katalog uygulama profili” tarafından tanımlanır. OGC, CSW katalog servisinin bilgi modelini tanımlamak için, 2004 yılına eBRIM profilini (OGC, 2004), 2005 yılında ise ISO 19115/19119 profilini (OGC, 2005a) geliştirmiştir. CSW’nin bilgi modelleri ile ilgili detaylar, uygulama profillerinin açıklandığı ileriki bölümlerde ele alınacaktır.

Katalog arayüzleri: CSW katalog servisleri, CSW belirtimi tarafından tanımlanan arayüzleri gerçekleştirirler. CSW belirtimi, bir CSW katalog servisinin gerçekleştirmesi gereken 5 arayüz (OGC_Service, Discovery, Manager, BrokeredAccess, Session) tanımlamıştır. “OGC Servis” (OGC_Service) ve “Bulma” (Discovery) arayüzü, tüm CSW katalog servisleri tarafından gerçekleştirilmesi zorunlu olan arayüzlerdir. Bir CSW katalog servisinin gerçekleştirebileceği arayüzler, katalog servisinin kullandığı uygulama protokolüne bağlı olarak değişir. Örneğin,

HTTP protokolü “durumsuz” (stateless) bir protokol olduğu için, bu protokolü kullanan bir CSW katalog servisi “Oturum” (Session) arayüzünü gerçekleştiremez. HTTP protokolünü kullanan bir CSW katalog servisi, OGC Servis ve Bulma arayüzlerinin yanı sıra, seçmeli olan “Yönetim” (Manager) arayüzünü de gerçekleştirebilir. CSW belirtimi, söz konusu arayüzlerin Z39.50, CORBA/IIOP ve HTTP protokollerindeki gerçekleştirmelerini tanımlamaktadır. Arayüzler ve içerdikleri operasyonlar, farklı protokolleri kullanan katalog servislerinde farklı isimlerle adlandırılırlar. Örneğin, HTTP protokolünü kullanan CSW gerçekleştirmelerinde, “Yönetim” arayüzü, “Yayınlama” (Publication) arayüzü olarak adlandırılmaktadır.

Bulma arayüzü: Bir CSW katalog servisinde arama yapmak için CSW’nin “Bulma” (Discovery) arayüzü kullanılır. Bulma arayüzü, biri seçmeli olan dört operasyon (*DescribeRecord*, *GetRecords*, *GetRecordById*, *GetDomain*) içerir. Katalog servisini sorgulamak ve meta veri kayıtlarını elde etmek için kullanılan temel operasyon, “*GetRecords*” operasyonudur. *GetRecords* operasyonu, CSW belirtimi tarafından tanımlanan CQL (Common Catalogue Query Language) veya OGC Filter (OGC, 2005c) sözdizimine uygun olarak yazılan bir sorgu ifadesi içerir. Sorgu ifadesi, katalog servisinde bilgi kaynaklarını tanımlamak için kullanılan meta veri sınıflarının öznelikleri kullanılarak yazılır. Bu nedenle, kullanıcının sorgusunu yazabilmesi için katalogun bilgi modelini bilmesi gerekmektedir. Kullanıcı, katalog servisinin bilgi modelini bilmiyorsa, Bulma arayüzünün “*DescribeRecord*” operasyonunu kullanarak, katalogdan bilgi modelinin şema tanımını alabilir.

GetRecords operasyonu, “*DistributedSearch*” olarak adlandırılan seçmeli bir parametre içermektedir. *DistributedSearch* parametresi, kullanıcının bir CSW katalog servisine gönderdiği sorgunun, sorgunun gönderildiği katalog servisi eğer bir katalog federasyonuna üye ise, federasyon içerisindeki diğer katalog servislerine gönderilip gönderilmeyeceğini belirtmektedir. OGC, CSW belirtimi ve katalog uygulama profilleri ile “bir CSW katalog servisi hangi arayüzleri gerçekleştirmelidir?” ve “katalog servisinin bilgi modeli ne olmalıdır?” gibi sorulara yanıt vermektedir. OGC, katalog servisleri arasında federasyonun nasıl kurulacağı ve sorgu yönlendirmenin nasıl yapılacağı gibi konuları, uygulama profillerini gerçekleştiren katalog sağlayıcılarına bırakmıştır.

OGC, bir kullanıcının, bir sorgu üzerinde değişiklik yapmadan ve katalog servislerinin bilgi modellerinin detaylarını bilmeden, aynı sorguyu kullanarak tüm CSW katalog servislerinde arama yapabilmelerini sağlamak için “Temel Sorgulanabilir Öznelikleri” (TSÖ) geliştirmiştir (OGC, 2005b). TSÖ’leri kullanarak katalogda arama yapacak olan bir kullanıcının, “*DescribeRecord*” operasyonunu kullanarak katalogun bilgi modelini öğrenmesine gerek yoktur. Kullanıcının, TSÖ’leri kullanarak yazdığı sorguyu, *GetRecords* operasyonu ile bir katalog servisine göndermesi yeterlidir. TSÖ’ler, kullanıcıların, özellikle farklı bilgi modellerini gerçekleştiren katalog servislerinin oluşturduğu bir federasyon içerisinde arama yapmaları durumunda önem kazanmaktadır. Bir kullanıcının, federasyon içerisindeki bir katalog servisinin bilgi modeline göre yazdığı sorgu, farklı bir bilgi modelini gerçekleştiren katalog servisi tarafından yürütülemez. Dolayısıyla, TSÖ’ler katalog servislerinin birlikte işlerliğini sağlarlar.

Yayınlama arayüzü: Sağlayıcılar, sahip oldukları bilgi kaynaklarını tanımlayan meta verileri, yayınlama arayüzünü kullanarak CSW katalog servislerine yayımlarlar. Yayınlama arayüzü, seçmeli olan “*Transaction*” ve zorunlu olan “*Harvest*” operasyonlarına sahiptir. *Transaction* operasyonu, meta verilerin bir CSW katalog servisine sağlayıcılar tarafından yayınlanmasına olanak sağlar. Sağlayıcılar, *transaction* operasyonunu kullanarak kataloglardaki meta veri kayıtlarını güncelleyebilir veya silebilirler. Meta verisini bir CSW katalog servisine yayınlamak isteyen bir sağlayıcı, *transaction* operasyonunu kullanarak katalog servisine XML formatında bir istek mesajı gönderir. İstek mesajı, sağlayıcı tarafından katalogun bilgi modeline uygun olarak yazılan meta verileri içerir. CSW katalog servisi, gelen istek mesajını okur ve meta verileri kataloga kaydeder.

Harvest operasyonu ise meta verilerin bir CSW katalog servisi tarafından otomatik olarak bir sunucudan alınmasını ve kataloga kaydedilmesini sağlar. *Harvest* operasyonunu kullanarak meta verilerini bir CSW’ye yayınlamak isteyen bir sağlayıcı, ilk olarak bir meta veri editörü kullanarak sahip olduğu bilgi kaynaklarını tanımlayan XML formatında meta veri dosyaları üretir. Sağlayıcı daha sonra katalog servisine bir istek mesajı (*HarvestRequest*) gönderir. İstek mesajı, meta veri dosyalarına erişim adresini, meta veri standardının adını ve sağlayıcı tarafından belirtilen bir zaman aralığını içerir. Katalog servisi, kullanıcı tarafından belirtilen adrese bağlanır ve meta veri dosyalarını alır, bir XML parser kullanarak meta verileri dosyalardan okur ve kataloga kaydeder. Katalog servisi, sağlayıcı tarafından belirtilen zaman aralığında, ilgili adrese yeniden bağlanarak katalog kayıtlarını günceller.

Diğer arayüzler: OGC servis modeline göre tüm OGC Web servisleri “*GetCapabilities*” operasyonunu gerçekleştirmek zorundadır. Bu operasyon, “OGC Servis” arayüzü içerisinde yer alır. *GetCapabilities* operasyonu, kullanıcılara, bir CSW katalog servisinin sorgu yeteneklerini, servisin gerçekleştirebildiği operasyonları ve bu operasyonların parametrelerini gösteren ve “yetenekler dökümanı” (capabilities document) olarak adlandırılan bir XML dokümanı döndürür. Kullanıcı, bu XML dokümanını inceleyerek katalog servisinin neler yapabileceği (yetenekleri) hakkında bilgi sahibi olur.

CSW katalog servisleri, katalog içeriğindeki değişiklikleri kullanıcılara otomatik olarak bildiren bir “olay bildirme” (event notification) arayüzüne sahip değillerdir. Katalog uygulama profilleri, CSW katalog servislerine bilgi modeli

desteğinin yanı sıra arayüz desteği de sağlayabilirler. Örneğin, CSW-ebRIM profilinin ilk sürümünde (OGC, 2004), ebRIM bilgi modelini gerçekleştiren CSW katalog servislerinin “Bulma” arayüzü, “WRS-Retrieval” olarak adlandırılan seçmeli bir arayüz ile genişletilmişti. CSW-ebRIM profilinin ikinci sürümünde ise, “Bulma” arayüzüne “GetRepositoryItem” olarak adlandırılan zorunlu bir operasyon eklenmiştir. CSW-ebRIM profilinin ikinci sürümü, ilk sürümü geçersiz kıldığı için “WRS-Retrieval” arayüzü hakkında bilgi verilmeyecektir. Ancak “GetRepositoryItem” operasyonu, CSW-ebRIM profilinin açıklandığı ileriki bölümde ele alınacaktır.

Katalog uygulama profili: Yukarıda da belirtildiği gibi CSW belirtimi, CSW katalog servislerinin bilgi modelini tanımlamamıştır. OGC, CSW katalog servisinin bilgi modelini tanımlamak için, 2004 yılına ebRIM profilini (OGC, 2004), 2005 yılında ise ISO19115/19119 profilini (OGC, 2005a) geliştirmiştir. Söz konusu profiller ile ilgili detaylar ileriki bölümlerde ele alınacaktır.

Anlamsal DesteK: Bir katalog servisinin anlamsal desteği, gerçekleştirdiği bilgi modelinin, bilgi kaynaklarını anlamsal olarak tanımlayabilme yeteneğine bağlıdır. CSW belirtimi, CSW katalog servisinin bilgi modelini tanımlamadığı için, burada CSW nin anlamsal desteği konusunda bir değerlendirme yapılamamaktadır. Bu nedenle, CSW katalog servisinin anlamsal desteği CSW’nin uygulama profilleri açıklanırken ele alınacaktır.

3.3.1. CSW-ebRIM Profili

ebRIM profili, OGC tarafından CSW katalog servislerine bilgi modeli desteği sağlamak için geliştirilmiştir. Profilin ilk sürümü 2004 yılında, ikinci sürümü (version 1.0.0) ise 2006 yılında yayınlanmıştır. ebRIM bilgi modeli ve CSW katalog servisi hakkında yukarıdaki bölümlerde detaylı bilgiler verilmişti. Dolayısıyla bu bölümde, CSW-ebRIM profilinin, özellikle ebRIM bilgi modeli ve CSW arayüzleri üzerinde yaptığı genişletmeler ele alınacaktır.

Bilgi modeli: CSW-ebRIM profiline uygun olarak geliştirilen bir CSW katalog servisi, ebRIM bilgi modelini gerçekleştirir. OGC, ebRIM bilgi modelinin konumsal bilgi toplumunun katalog servisi gereksinimlerini karşılayabilmesi için özelleştirilebileceği noktaları belirlemiş ve bu noktaları kullanarak bilgi modelinde bir takım genişletmeler yapmıştır (OGC, 2006). Söz konusu genişletmeler aşağıda maddeler halinde açıklanmıştır.

- ebRIM bilgi modelindeki nesne tipleri konumsal bilgi ihtiyaçları doğrultusunda genişletilmiştir. ebRIM bilgi modelindeki bütün sicil (registry) nesnelere, “objectType” olarak adlandırılan bir öznitelige sahiptir. *ObjectType* özniteliği, değerini ebRIM’in “ObjectType taksonomisi”nden almaktadır. *ExtrinsicObject* (EO) ve *ExternalLink* nesnelere dışındaki tüm nesnelere *objectType* özniteliklerinin değeri, nesne isimleri ile aynıdır. Örneğin, *Service* nesnesinin *objectType* özniteliğinin değeri yine “Service” dir. EO nesnesinin *objectType* özniteliğinin değeri ise, bu nesnenin tanımladığı içeriğin tipine bağlı olarak değişmektedir. Örneğin, EO nesnesi bir XML şema dokümanını tanımlıyorsa, *objectType* özniteliği “XML Schema” değerini alır. ebRIM’in *ObjectType* Taksonomisi, konumsal bilgi kaynaklarını tanımlayacak kadar geniş bir kategori içermemektedir. Örneğin, söz konusu taksonomide, ISO 19115 meta veri dosyalarını tanımlamak için kullanılacak bir obje tipi yoktur. Bu nedenle CSW-ebRIM profili, *ObjectType* Taksonomisine yeni obje tipleri eklemiştir.
- Konumsal bilgi kaynakları arasındaki ilişkileri temsil etmeye yönelik yeni ilişki tipleri eklenmiştir. Bunun nedeni, ebXML standardının konumsal olmayan, “e-iş” sektörüne yönelik olarak tasarlanmış olmasıdır. Örneğin ebRIM de, bir konumsal veri seti ile bu veriyi sunan OGC Web servisi arasındaki ilişkiyi tanımlamak için kullanılacak bir ilişki tipi yoktur.
- ISO 19119 Coğrafi Servis Taksonomisi ve DIGEST FACC “Detay Kodları” taksonomisi gibi, konumsal servis ve verilere yönelik olan ve ebXML bilgi modelinin desteklemediği yeni taksonomilerin kullanılabilmesi için gerekli destek sağlanmıştır.
- Farklı bilgi kaynağı yayıncılarının farklı *Slot* isimleri kullanmasının önlenmesi ve dolayısıyla birlikte işlerliğin sağlanabilmesi amacıyla standart *Slot* isimleri belirlenmiştir. Bu *Slot* isimleri için, Dublin Core meta veri terimleri esas alınmıştır.

Katalog arayüzleri: CSW-ebRIM profiline uygun olarak geliştirilen bir katalog servisi, CSW belirtimi tarafından tanımlanan arayüzleri gerçekleştirmek zorundadır. CSW-ebRIM profili, katalog servislerinin HTTP protokolünü kullanmalarını istemektedir. Bu nedenle katalog servisleri, CSW’nin “OGC Service”, “Yayınlama” ve “Bulma” arayüzlerini gerçekleştirebilirler. Bununla birlikte, ebRIM profili CSW’nin standart “Bulma” arayüzünü, gerçekleştirilmesi zorunlu olan “GetRepositoryItem” operasyonu ile genişletmiştir. İstemcilerin, bir katalog servisinin depo (repository) bileşeninde yer alan dokümanlara ulaşmalarını sağlayan *GetRepositoryItem* operasyonu, ebRS standardı tarafından tanımlanan bir operasyondur. ebXML katalog servislerinde, katalogun depo bileşeninde yer alan bir dokümanın, sicil tarafında EO nesnesi kullanılarak tanımlandığını ve EO nesnesi ile dokümanın aynı

id'ye sahip olduğunu hatırlayalım. *GetRepositoryItem* operasyonu, parametre olarak EO nesnesinin id'sini alır ve geriye EO nesnesi tarafından tanımlanan dokümanı döndürür.

Bulma arayüzü: CSW-ebRIM profilini gerçekleştiren bir katalog servisi, CSW belirtimi tarafından tanımlanan bulma arayüzünü destekler. Katalog servisi, Bulma arayüzünün *GetRecords* operasyonu kullanılarak sorgulanır. *GetRecords* operasyonu, ebRIM bilgi modelindeki meta veri sınıflarının öznitelikleri kullanılarak yazılan bir sorgu ifadesi içerir. CSW-ebRIM profili, sorgu ifadelerinin CQL ve OGC Filter sözdizimlerine ilave olarak XPath ve XQuery sorgu dilleri kullanılarak ta yazılmasına olanak sağlamaktadır. CSW-ebRIM profili, kullanıcıların katalog servislerinde içerik tabanlı sorgu yapmasına olanak sağlayan bir yöntem tanımlamaktadır. "Derin arama" (deep search) adı verilen bu yöntemde, kullanıcıların XPath sorgu dilini kullanarak katalog servisinde içerik tabanlı sorgular gerçekleştirebileceği belirtilmektedir. Kullanıcılar, katalog servisine gönderdikleri XPath sorgusu ile sadece "isOpaque" özniteliğinin değeri "false", "mimeType" özniteliğinin değeri "XML" olan EO nesnelere tarafından tanımlanan XML formatındaki dokümanları sorgulayabilirler. Derin arama yöntemini gerçekleştiren katalog servisleri, bu özelliklerini, yetenekler dökümanında belirtirler.

CSW-ebRIM profili, bu profili gerçekleştirecek olan bir katalog servisinin gerçekleştirmesi gereken saklanmış sorguları da tanımlamaktadır. Profil tarafından tanımlanan saklanmış sorgular, bir katalog servisinde kullanıcıların yaygın olarak kullanacağı düşünülen sorgulardan oluşmaktadır. ISO 19119 Coğrafi Servis taksonomisindeki kategorilere uyan Web servislerini bulmak için tasarlanan "findService" sorgusu, profil tarafından tanımlanan saklanmış sorgulara örnek olarak gösterilebilir. CSW belirtiminde, katalog servislerinin dağıtık sorgulamayı (DistributedSearch) destekledikleri belirtilmişti. Ancak ebRIM profili, dağıtık sorgulamanın nasıl yapılacağını tanımlayan bir öneri içermemekte ve bu özelliğin gerçekleştirimini katalog sağlayıcılarına bırakmaktadır.

Yayınlama arayüzü: CSW-ebRIM profilini gerçekleştiren OGC katalog servisleri, CSW nin yayınlama arayüzünü desteklerler. Yayınlama arayüzü, seçmeli olan *transaction* ve zorunlu olan *harvest* operasyonlarını içermektedir. Her iki operasyonun işleyişi, OGC CSW katalog servisinin yayınlama arayüzünün açıklandığı yukarıdaki bölümde ele alınmıştır.

Diğer arayüzler: CSW-ebRIM profilini gerçekleştiren OGC katalog servisleri, CSW nin "OGC Servis" arayüzünü gerçekleştirirler. OGC Servis arayüzü, tüm OGC Web servisleri tarafından gerçekleştirilmek zorunda olan *GetCapabilities* operasyonunu içerir. *GetCapabilities* operasyonu, kullanıcılara katalog servisinin yetenekler dökümanını döndürür. ebRIM profili, CSW katalog servisleri için ilave başka bir arayüz desteği içermemektedir.

Anlamsal destek: ebRIM bilgi modelinin anlamsal desteği, ebXML KS nin anlamsal desteğinin değerlendirildiği yukarıdaki bölümde açıklanmıştır. Ancak CSW-ebRIM profili, ebRIM bilgi modeli için ilave bir anlamsal genişletme içermemektedir.

3.3.2. CSW-ISO 19115/19119 Profili

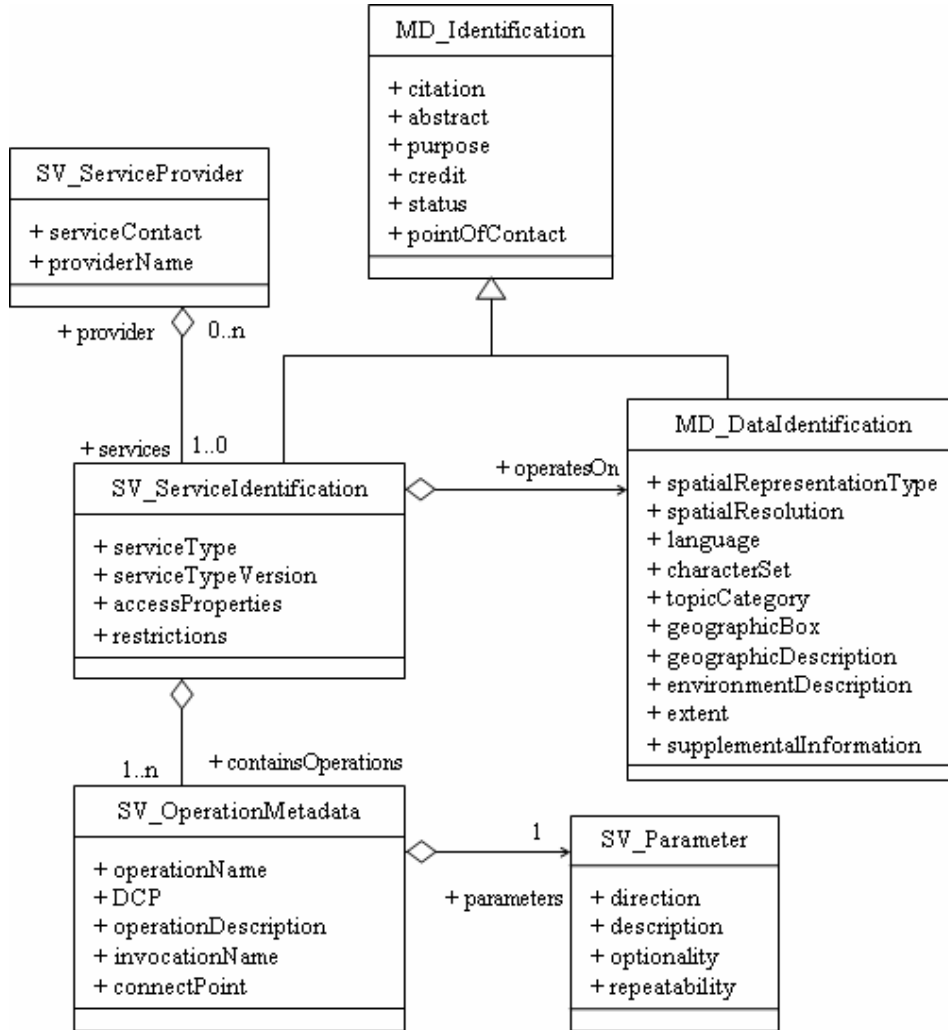
OGC, CSW belirtimi ile bir katalog servisinin gerçekleştirmesi gereken arayüzleri ve operasyonları tanımlamıştır. OGC, bugüne kadar yayınladığı CSW belirtimlerinde, katalog servisleri için bir bilgi modeli tanımlamamış ve katalog sağlayıcılarının da belirli bir bilgi modelini kullanmalarını şart koşmamıştır (OGC,2005b). Ancak OGC, "OWS-1" girişiminde (OGC, 2002), katalog bilgi modeli olarak ebRIM standardının kabul edildiği bildirilmesine ve 2004 yılında ebRIM profilinin yayınlanmasına rağmen, 2005 yılında katalog servisleri için yeni bir bilgi modeli tanımlayan ISO 19115/19119 profilini geliştirmiştir. CSW-ISO 19115/19119 profili, Almanya'nın North Rhine Westphalia eyaletinin KVA çalışmalarında, katalog servisi gerçekleştirimlerinde, bilgi modeli olarak ISO 19115 ve 19119 metaveri standartlarından geliştirilen bir uygulama profilinin kullanılmasından esinlenerek geliştirilmiştir.

Bilgi modeli: CSW-ISO 19115/19119 profiline uygun olarak geliştirilen CSW katalog servisleri, bilgi modeli olarak ISO 19115 (ISO, 2003) ve ISO 19119 (ISO, 2001) metaveri standartlarını gerçekleştirirler. ISO 19115 konumsal veriler, ISO 19119 ise Web servisleri ile ilgili meta verileri tanımlamak için geliştirilen standartlardır.

Bilgi kaynağı türü: CSW-ISO 19115/19119 profili, ebRIM bilgi modelinde olduğu gibi her türlü elektronik dokümanı tanımlamaya olanak sağlayan genel bir bilgi modeline sahip değildir. Profil, sadece konumsal verilerin ve Web servislerinin tanımlanmasına olanak sağlayan bir bilgi modeli sunmaktadır. Örneğin bilgi modeli, bir SLD dosyasını veya GML formatından SVG formatına dönüşüm yapmak için kullanılan bir XSLT dosyasını tanımlamak için kullanılamaz.

Anlatım gücü: ISO 19115 meta veri standardı, konumsal verilerle ilgili meta verileri tanımlayan 12 adet temel meta veri sınıfı ve yaklaşık 300 adet öznitelik içerir. ISO 19119 ise, Web servislerini tanımlayan 4 adet meta veri sınıfı içermektedir (Şekil 4). ISO 19115 ve 19119, nesne-yönelimli bir tasarıma sahip değildir. Ancak meta veri sınıfları arasındaki ilişkileri tanımlamak için, genelleştirme (generalization) ve bir araya getirme (aggregation) gibi nesne yönelimli modellemede kullanılan kavramları kullanırlar. Örneğin, ISO 19115 meta veri standardında yer alan ve

konumsal verilerle ilgili meta verileri tanımlamak için kullanılan ana sınıf olan “MD_Metadata” sınıfı, 11 adet meta veri sınıfını içerir. Bu sınıflardan sadece “MD_Identification” sınıfı meta veri tanımı içerisinde yer alması zorunlu olan sınıftır. Diğer sınıflar ise seçmelidir. Ayrıca MD_Identification sınıfı, “MD_DataIdentification” ve “SV_ServiceIdentification” sınıflarının her ikisinin de üst sınıfı yani daha genel halidir (Şekil 4). Her iki meta veri standardında da, eBRIM bilgi modelinde olduğu gibi meta veri sınıfları arasındaki ilişkileri tanımlamak için kullanılan Association sınıfı gibi bir sınıf yoktur.



Şekil 4. ISO 19119 da servis meta verisini tanımlamak için kullanılan meta veri sınıfları (ISO, 2003b).

Genişletilebilirlik: ISO 19115 ve 19119 meta veri standartları, genişletilebilir bir bilgi modeli sunmaktadır. Her iki standart da, kullanıcıların bilgi modeline yeni meta veri sınıfları eklemesine veya bir meta veri özneliğinin alabileceği değerlerin genişletilmesine olanak sağlamaktadır. CSW-ISO 19115/19119 profili, ISO 19115 standardında yer alan meta veri sınıflarında bir genişletme yapmamıştır. Ancak profil, ISO 19119 bilgi modelinde bir takım genişletmeler yapmış ve bilgi modeline iki yeni sınıf eklemiştir. CSW-ISO 19115/19119 profili, Web servislerini, “gevşek bağlı” (loosely coupled), “sıkı bağlı” (tightly coupled) ve “karışık bağlı” (mixed coupled) Web servisleri olarak üçe ayırmaktadır. Söz konusu bağlılık düzeyleri, bir Web servisi ile bir konumsal veri seti arasındaki ilişkiyi temsil etmektedir. Sıkı bağlı Web servislerinde, bir Web servisi belirli bir veri seti ile ilişkilidir. Örneğin, bir OGC WFS servisi sadece kadastro verisini sunuyor ise bu sıkı bağlı bir Web servisidir. Gevşek bağlı Web servislerinde ise bir Web servisi belirli bir veri seti ile ilişkili değildir. Ancak servis, belirli bir veri tipi ile ilişkili olabilir. Örneğin bir Web servisi, sadece nokta tipine sahip olan verileri sunuyor ise söz konusu servis, gevşek bağlı bir servistir. Karışık bağlı Web servislerinde ise bir Web servisi hem belirli bir veri seti ile hem de belirli bir veri tipi ile ilişkili olabilir. CSW-ISO 19115/19119 profili, Web servisleri ile konumsal veri setleri arasındaki söz konusu bağlılık ilişkilerini temsil edebilmek için bilgi modeline, “CSW_ServiceIdentification” ve “CSW_CoupledResource” sınıflarını eklemiştir.

Katalog arayüzleri: CSW-ISO 19115/19119 profiline uygun olarak geliştirilen CSW katalog servisleri, CSW belirtimi tarafından tanımlanan OGC Service, yayınlama ve bulma arayüzlerini gerçekleştirebilirler. CSW-ISO 19115/19119 profili, katalog servislerini gerçekleştirdikleri arayüzlere göre “salt okunur” (read-only) ve “işlevsel” (transactional) katalog servisleri olarak ikiye ayırmaktadır. Salt okunur (CSW) katalog servisleri, sadece OGC servis

ve bulma arayüzünü gerçekleştirirler. İşlevsel katalog servisleri (CSW-T) ise, bu arayüzlere ilave olarak yayınlama arayüzünü de gerçekleştirirler. OGC servis ve yayınlama arayüzlerinde yer alan operasyonlar, yukarıda CSW'nin katalog arayüz desteğinin anlatıldığı bölümde açıklanan şekilde yürütülürler. CSW-ISO 19115/19119 profili, CSW arayüzlerinde bir genişletme yapmamıştır.

Bulma arayüzü: CSW-ISO 19115/19119 profilini gerçekleştiren bir katalog servisi, CSW belirtimi tarafından tanımlanan bulma arayüzünü gerçekleştirir. Ancak bu profili gerçekleştiren katalog servislerinde, kullanıcılar sorgularını ISO 19115 ve ISO 19119 meta veri standartlarında yer alan meta veri sınıflarının özniteliklerini kullanarak yazarlar.

Yayınlama arayüzü: CSW-ISO 19115/19119 profili, katalog servislerini gerçekleştirdikleri arayüzlere göre “salt okunur” ve “işlevsel” katalog servisleri olarak ikiye ayırmaktadır. Yayınlama arayüzü, sadece işlevsel katalog servisleri (CSW-T) tarafından gerçekleştirilmektedir. Yayınlama arayüzü, seçmeli olan “*transaction*” ve zorunlu olan “*harvest*” operasyonlarına sahiptir. Her iki operasyonun işleyişi bölüm 3.3. de açıklanmıştır.

Diğer arayüzler: CSW-ISO 19115/19119 profiline uygun olarak geliştirilen CSW katalog servislerinin, CSW belirtiminde belirtilen arayüzler dışında, örneğin “olay bildirme” arayüzü gibi farklı bir arayüz destekleri yoktur.

Katalog uygulama profili: CSW-ISO 19115/19119 profili, katalog servisi sağlayıcılarına özelleştirilebilir bir bilgi modeli sunmaktadır. ISO 19115 meta veri standardı, 12 adet temel meta veri sınıfı ve yaklaşık 300 adet öznitelik içermektedir. Bu sınıfların ve özniteliklerin büyük bir çoğunluğu seçmelidir. Bir katalog servisi sağlayıcısının, bu meta veri sınıflarının veya özniteliklerin tamamına ihtiyacı olmayabilir. Örneğin, sadece vektör formatta konumsal veri sunan bir organizasyona ait katalog servisinde, “*MD_SpatialRepresentation*” sınıfının bir alt sınıfı olan “*MD_GridSpatialRepresentation*” sınıfının bilgi modelinde yer almasına gerek yoktur. Bu nedenle sağlayıcılar, bilgi modelini ihtiyaçları doğrultusunda özelleştirebilirler. Ancak bilgi modelinin bu şekilde her kurum ya da her uygulama için özelleştirilmesi, katalog servisleri arasında birlikte işlerlik problemlerine sebep olabilir.

Anlamsal destek: CSW-ISO 19115/19119 profili tarafından tanımlanan bilgi modeli, anlamsal bir destek içermemektedir. Bu konuyla ilgili olarak gerek OGC de gerekse akademik çevrelerde bir çalışmaya rastlanamamıştır. Bilgi modelini oluşturan ISO standartlarında da bu konuyla ilgili bir bilgi yer almamaktadır.

4. SONUÇ

Konumsal Veri Altyapıları (KVA), kamu kurumları, özel sektör, yerel yönetimler ve konumsal veri ile iş yapan bütün kesimler arasında “birlikte işlerliği” sağlayarak, hızlı, kaliteli ve ekonomik iş ya da çözümler üretmeye olanak tanır. KVA'lar aslında bir birlikte işlerlik altyapısıdır. Birlikte işlerlik, genel olarak farklı dil ya da kavramlar kullanan uygulamaların birbirleri ile konuşabilmesi olarak tanımlanabilir. Farklı programlama dilleri kullanılarak geliştirilen, ağ üzerinde farklı yerlerde bulunan ve farklı platformlara sahip bilgisayarlar üzerinde koşan uygulamaların, belirli görevleri yerine getirebilmek için, birlikte işleyebilmelerine olanak sağlayan çeşitli sistemler ve yazılım mimarileri geliştirilmiştir. Şu an oldukça popüler ve yaygın olan yazılım mimarisi, “Servis Yönelimli Mimari” (SyM) dir. Web servisleri, SyM'yi gerçekleştirenin en iyi ve şu anki en popüler yolu olarak kabul edilmektedir. Bununla birlikte, bir KVA'nın Web servisleri ile başarılı bir gerçekleştirimi için üzerinde durulması gereken birçok sorun mevcuttur. Bu sorunların bir kısmı, katalog servisleri ile ilgilidir. UKVA gibi geniş ölçekli bir KVA gerçekleştiriminde çok sayıda servis ve bunların sağlayıcıları söz konusu olacaktır. Bu durumda bir istemci gereksinimlerini karşılayacak özelliklerdeki bir ya da daha çok servisi nasıl bulacaktır? Bunun ne gibi boyutları vardır? Bu bağlamda katalogların önemi nedir? Servisler merkezi bir katalog da mı depolanacaktır yoksa birden çok katalog mu olmalıdır? Birden çok ve farklı katalog kullanılması durumunda birlikte işlerlik sağlanabilecek midir? Bütün bu sorulara yönelik ne gibi çözümler ve araçlar mevcuttur ve bunlar bir çerçeve de değerlendirilebilir mi? gibi sorular bu çalışmada ele alınan konunun özünü teşkil etmiştir.

Anılan sorunların basitleştirici ve bütüncül bir yaklaşımla ele alınmasına yardımcı olacak bir çatıya yönelik olarak kataloglama bağlamında bir takım ölçütler belirlenebilir mi? noktasından hareketle bu çalışmada bir dizi ölçüt belirlenmiştir. OASIS ortaklığı tarafından geliştirilen ve standart olarak kabul edilen UDDI ve ebXML Katalog Servisi ile OGC konsorsiyumu tarafından geliştirilen CSW katalog servisi ve bu servisin ISO 19115/19119 ve ebRIM uygulama profilleri belirlenen ölçütlere göre değerlendirilmiştir. Değerlendirme sonucunda, zengin bir arayüz desteğine sahip olması, her türlü bilgi kaynağını tanımlamaya olanak sağlaması, farklı kullanıcı gruplarının isteklerini karşılayabilmek için özelleştirilebilmesi, bilgi kaynağı tanımlamalarını anlamsal olarak zenginleştirmek için genişletilebilmesi ve nesne yönelimli bir bilgi modeline sahip olması nedeniyle, KVA gerçekleştirimlerinde ebRIM bilgi modelini gerçekleştiren ebXML KS veya CSW-ebRIM profilini gerçekleştiren OGC CSW katalog servislerinin kullanılmasının uygun olacağı ortaya çıkmıştır.

KAYNAKLAR

- Akıncı, H.**, 2006. *Konumsal Veri Altyapılarının Web Servisleri ile Gerçekleştirilmesi: Mevcut Durum Analizi ve Gelecek Yönelimlerinin Belirlenmesi*, Doktora Tezi, KTÜ Fen Bilimleri Enstitüsü, Trabzon.
- Batini, C., Ceri, S., Navathe, S.**, 1992. *Conceptual Database Design: An Entity-Relationship Approach*, The Benjamin/Cummings Publishing Company, Inc., California, USA.
- Berners-Lee, T., Hendler, J., Lassila, O.**, 2001. *The Semantic Web*, Scientific American, 284, 4, 34–43.
- Cerami, E.**, 2002. *Web Services Essentials*, O'Reilly & Associates, Inc., Sebastopol, USA.
- Colan, M.**, 2004. *Service-Oriented Architecture expands the vision of Web services, Part I: Characteristics of Service-Oriented Architecture*, <http://www-128.ibm.com/developerworks/webservices/library/ws-soaintro.html>
- Di, L., Zhao, P., Yang, W., Yue, P.**, 2006. *Ontology-driven Automatic Geospatial-Processing Modeling based on Web-service Chaining*, 6th Annual NASA Earth Science Technology Conference, Adelphi, Maryland, USA.
- Dogaç, A., Cingil, İ., Laleci, G., Kabak, Y.**, 2002a. *Improving the Functionality of UDDI Registries through Web Service Semantics*, 3rd VLDB Workshop on Technologies for E-Services (TES-02), Hong Kong, China.
- Dogaç, A., Laleci, G., Kabak, Y., Cingil, İ.**, 2002b. *Exploiting Web Service Semantics: Taxonomies vs. Ontologies*, IEEE Data Engineering Bulletin, 25, 4, 10–16.
- Dogac, A.**, 2003. *A Tutorial on Exploiting Semantic of Web Services through ebXML Registries*, eChallenges 2003, Bologna, Italy.
- Dogac, A., Kabak, Y., Laleci, G. B.**, 2003. *A Semantic-Based Web Service Composition Facility for ebXML Registries*, 9th International Conference of Concurrent Enterprising, Dipoli Congress Center, Espoo, Finland.
- Dogac, A., Kabak, Y., Laleci, G. B.**, 2004. *Enriching ebXML Registries with OWL Ontologies for Efficient Service Discovery*, 14th International Workshop on Research Issues on Data Engineering: Web Services for E-Commerce and E-Government Applications (RIDE'04), Boston, USA.
- Dogac, A., Kabak, Y., Laleci, G. B., Mattocks, C., Najmi, F., Pollock, J.**, 2005. *Enhancing ebXML Registries to Make them OWL Aware*, Distributed and Parallel Databases, 18, 9–36.
- Dustdar, S., Treiber, M.**, 2005. *A View Based Analysis on Web Service Registries*, Distributed and Parallel Databases, 18, 2, 147-171.
- FGDC**, 1998. *Content Standard for Digital Geospatial Metadata*, Federal Geographic Data Committee, FGDC-STD-001-1998.
- ISO**, 2001. *Geographic information — Services*, ISO/DIS 19119, ISO TC 211/WG 4.
- ISO**, 2003. *Geographic information — Metadata*, ISO/FDIS 19115, ISO TC 211.
- Manes, A. T.**, 2002. *Standardizing Web Services*, Upgrade Magazine, www.siiia.net, http://www.siiia.net/upgrade/archive/0809_02/manes.pdf, 22-25.
- Meyer, B.**, 1988. *Object-Oriented Software Construction*, First Edition, Prentice-Hall, Inc. Upper Saddle River, NJ, USA.
- Najmi, F.**, 2005. *Registry Capability Comparison Matrix*, http://ebxmlrr.sourceforge.net/tmp/Registry_Capability_Matrix.html, 26 Aralık 2005.
- Newcomer, E.**, 2002. *Understanding Web Services XML, WSDL, SOAP and UDDI*, Addison-Wesley, Indianapolis, USA.
- OASIS**, 2004. *Universal Description, Discovery and Integration (UDDI)*, Version 3.0.2, <http://uddi.org/pubs/uddi-v3.0.2-20041019.htm>, 28 Ağustos 2005.
- OASIS**, 2005a. *ebXML Registry Services and Protocols*, Version 3.0, OASIS Standard.

- OASIS**, 2005b. *ebXML Registry Information Model*, Version 3.0, OASIS Standard.
- OGC**, 1999. *Catalogue Interface Implementation Specification*, Version: 1.0, OGC 00–034, OGC Implementation Specification.
- OGC**, 2002. *OWS–1 Registry Service (WRS)*, OGC 02-050r5, Version: 0.7.2.
- OGC**, 2004. *OGC Catalogue Services – ebRIM (ISO/TS 15000–3) profile of CSW*, OGC 04-017r1, Version: 0.9.1, Discussion Paper.
- OGC**, 2005a. *OpenGIS Catalogue Services Specification 2.0 - ISO19115/ISO19119 Application Profile for CSW 2.0*, OGC 04-038r2, Version: 0.9.3, OGC Recommendation Paper.
- OGC**, 2005b. *OGC Catalogue Services Specification*, OGC Implementation Specification, Version 2.0 with corrigendum (OGC 04-021r3), Eds.: D. Nebert, A. Whiteside, Open Geospatial Consortium Inc.
- OGC**, 2005c. *Filter Encoding Implementation Specification*, OGC 04–095, version 1.1.0, OGC Implementation Specification.
- OGC**, 2006. *OGC Catalogue Services – ebRIM (ISO/TS 15000-3) profile of CSW*, OGC 05-025r3, Version: 1.0.0, OGC Application Profile.
- OMG**, 2005. *Unified Modeling Language: Infrastructure*, version 2.0, formal/05-07-05, <http://www.omg.org/cgi-bin/doc?formal/05-07-05>, 29 Haziran 2006.
- Paolucci, M., Kawamura, T., Payne, T., Sycara, K.**, 2002a. *Importing the Semantic Web in UDDI*, International Workshop on Web Services, E-Business, and the Semantic Web (WES 2002), Toronto, Canada.
- Paolucci, M., Kawamura, T., Payne, T., Sycara, K.**, 2002b. *Semantic Matching of Web Services Capabilities*, The Semantic Web - ISWC 2002: First International Semantic Web Conference, Sardinia, Italy.
- Paolucci, M., Soudry, J., Srinivasan, N., Sycara, K.**, 2004. *A Broker for OWL-S Web Services*, First International Semantic Web Services Symposium, Southampton, UK.
- Ryman, A.**, 2000. *Understanding Web Services*, http://www7b.software.ibm.com/wsdd/library/techarticles/0307_ryman/ryman.html, 27 Haziran 2003.
- Saltor, F., Castellanos, M., Garcia-Solaco, M.**, 1991. *Suitability of data models as canonical models for federated databases*, *SIGMOD Record*, 20, 4, 45–48.
- Sivashanmugam, K., Verma, K., Sheth, A. P.**, 2004. *Discovery of Web Services in a Federated Registry Environment*, Proceedings of the IEEE International Conference on Web Services (ICWS'04), volume 00, 270-278.
- ShaikhAli, A.**, 2003. *UDDIe: An Extended Registry for Web Services*, Service Oriented Computing: Models, Architectures and Applications, SAINT-2003 IEEE Computer Society Pres, Orlando Florida, USA.
- UDDI.org**, 2000. *UDDI Technical White Paper*, Ariba Inc., IBM Corp., and Microsoft Corp., http://www.uddi.org/pubs/Iru_UDDI_Technical_White_Paper.pdf, 24 Temmuz 2003.
- W3C**, 2004. *Web Services Architecture*, W3C Working Group Note, <http://www.w3.org/TR/2004/NOTE-ws-arch-20040211/>, 11 Kasım 2005.
- Wiederhold, G.**, 1992. *Mediators in the Architecture of Future Information Systems*, *IEEE Computer*, 25, 3, 38-49.
- Vinoski, S.**, 2002. *Web Services Interaction Models, Part I: Current Practice*, *IEEE Internet Computing*, 6,3,89-91.
- Yu, L.**, 2006. *Understanding UDDI's tModel*, <http://www.codeproject.com/soap/understandingTModels.asp>, 17 Şubat 2006.