

E-Fatura Yapısal ve Anlamsal Kontrol Yazılımının Performans Analizi

Salih Bayar^{1,2} and Mehmet Yasin Akpınar¹

¹ Boğaziçi Üniversitesi, Bilgisayar Mühendisliği,
Bebek, İstanbul

² İdea Teknoloji Çözümleri,

Sun Plaza BBDO Blok Dereboyu Cd. Bilim Sk No:5 34398 Maslak/İstanbul
{yasin.akpınar, salih.bayar}@boun.edu.tr,
{salih.bayar}@ideateknoloji.com.tr

Özetçe.

E-fatura sisteminin zorunlu hale getirilmesiyle birlikte kullanımı oldukça yaygınlaşmıştır. Bu da beraberinde yazılımsal bir yük ortaya çıkarmaktadır. e-Fatura Schematron kurallarının kontrolü de bu yükün önemli bir kısmını oluşturmaktadır. Bu çalışmada e-Fatura Schematron kurallarının tek bir XSLT dosyasında çalıştırılması ile her bir kuralın farklı XSLT dosyalarında çalıştırılma performansları kıyaslanmıştır. Elde edilen sonuç doğrudan bu işlem süresinin minimum sürede tamamlanmasını sağlayacak olup hem mükellef hem de entegratör firmalar için hayati önem taşımaktadır. Bunun için örnek bir e-Fatura UBL dosyası ile 12 kural içeren bir XSLT dosyası Saxon kütüphanesi kullanılarak Schematron kontrolü gerçekleştirilmiştir. Bu işlem hem kuralların hepsi tek bir XSLT dosyası içerisindeyken hem de kurallar ayrıştırılıp 12 ayrı XSLT dosyası halindeyken yapılmıştır. Ayrıca her iki durum için de template yapısı kullanılarak toplamda 4 durum göz önüne alınmıştır. Testlerin anlık bilgisayar yükünden etkilenmemesi için her test 100'er defa tekrarlanıp ortalama hesabı yapılmıştır.

Anahtar Kelimeler: e-Fatura, XSD, Schematron, HTML, JSOUP, Saxon

1 Giriş

E-Fatura uygulaması, hem Türkiye' de hem de diğer ülkelerde kullanılmakta olup, elektronik belge biçiminde düzenlenen faturaların, taraflar arasında dolaşımını güvenli ve sağlıklı biçimde sağlamaktadır. Türkiye' de 397 nolu Vergi Usul Kanunu (VUK) Genel Tebliği ile 2010 yılında e-Fatura uygulaması hizmete alınmıştır. Nisan 2014 itibari ile 421 no.lu VUK Genel Tebliğinde belirtilen firmaların kullanması zorunludur. E-Fatura aynı zamanda, ilgili yetkili kurum ve merciler için mükelleflerden vergi toplama kolaylığı sağlayıp, alıcılar ve satıcılar arasındaki fatura kesme ve alma işlemlerini hızlandırma ve fatura işlemlerindeki maliyetleri düşürme amaçlı bir uygulamadır [1].

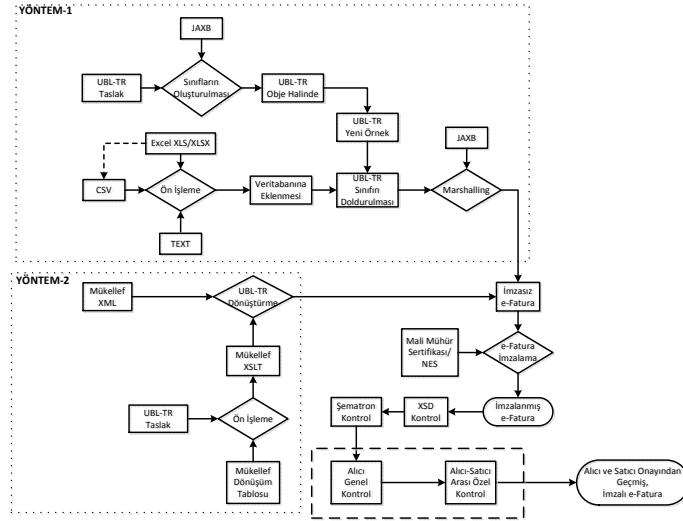
E-Fatura uygulaması kapsamında, alıcı ve fatura kesen taraf e-Fatura sisteminde olmalıdır. E-Fatura kapsamında 17 Haziran 2016 itibari ile 51076 kayıtlı

kullanıcı bulunmakta olup, Türkiye’de yıllık minimum 5.8 milyar adet e-Fatura beklentisi vardır [2].

2010 yılında Türkiye genelinde düzenlenen e-Fatura sayısı 8 bin civarındayken, bu rakam 2016 yılı itibari ile 113 milyon civarındadır. Artan e-Fatura sayısı ve tutarı, e-Fatura denetim ve kontrollerinin de zorunluluğunu beraberinde getirmiştir. Bu çalışma kapsamında e-Fatura’ nın denetiminde GİB tarafından belirlenen zorunlu kuralların ve GİB tarafından henüz belirtilmemiş olup, mükellefleri güvenli alışverişe yönlendiren, e-Fatura iptal sayı ve sürelerini asgari seviyeye indiren kontrollerde kullanılan yöntemler ve teknikler anlatılmaktadır.

Bu çalışmanın devamında Bölüm 2’ de e-Fatura’ nın oluşturma adımları ve oluşum sonrası yapılan zorunlu, satıcı ve alıcıyı korumaya yönelik kuralların işleme yöntemlerinden bahsedilmektedir. Bölüm 3’ de e-Fatura oluşumu sonrası yapılan tetkiklerde kullanılacak farklı yöntemler üzerinde durulmuş olup, hangi yöntemin daha etkin olduğundan bahsedilerek, ilgili performans testleri detaylı bir şekilde yapılmıştır. Son olarak Bölüm 4’ de çalışmadan çıkan sonuçlar ele alınmıştır.

2 E-Fatura Oluşturma



Şekil 1. E-Fatura Oluşturma Adımları

E-Fatura oluşturulurken vergi mükelleflerinden gelen e-Fatura verisinin yapısına göre genel olarak iki farklı yöntem kullanılmaktadır. Bu iki farklı yöntem Şekil 1’ de verildiği üzere, vergi mükelleflerinin ERP sistemlerinden gönderdikleri e-Fatura’ nun oluşturuluş şekline göre farklılık göstermektedir. Şekil 1’ de gösterildiği gibi,

eğer e-Fatura TXT, CSV, Excel dosyası şeklindeyse *Yöntem-1*, aksi durumda yani e-Fatura belirgin bir XML formatında verilirse *Yöntem-2* kullanılmaktadır. Bu yöntemler, önceki çalışmamız olan [4]’ de zaten detaylı olarak incelenmişti.

Bu çalışma kapsamında yapılan temel farklılık e-Fatura oluşumundan sonra yapılan XSD ve Schematron kontrolleri sonrası, imzalı UBL-TR formatındaki e-Fatura verisinin alıcının önceden belirlemiş olduğu genel kontrollerden ve alıcı-satıcı arası yine önceden tanımlanmış, e-Fatura iptalini neredeyse gereksiz hale getiren kuralların kontrolleri de verilmiştir. Bu kural kontrolleri Şekil 1’ de kesikli çizgi ile gösterilmektedir.

Şekil 1’ de kesikli çizgi ile belirtilmiş olan, alıcının genel kurallarını içeren ve alıcı-satıcı arası özel kuralları içeren kontroller sayesinde e-Fatura iptal sayılarının asgari seviyeye indirgenmesi hedeflenmektedir.

Şekil 1’ den de anlaşılacağı üzere, UBL-TR formatındaki e-Fatura verisi, XSD ve Schematron kontrolleri dahil arka arkaya toplam dört farklı, birbirinden bağımsız kontrollerden geçmektedir. Hal böyle olunca, e-Fatura verisinin işleme süresinin de hesaba katılması gerekmektedir. Arka arkaya yapılan bu kontrollerin en etkin ve hızlı şekilde yapılması için Bölüm 3’ de ilgili performans testleri verilmiştir.

3 Performans Testleri ve Deneyimler

Senaryo Adı	Açıklaması
<i>All_Original</i>	Kontrolü yapılacak bütün kurallar bir xslt dosyası içerisinde listelenmiştir. Template kullanılmamıştır ve dolayısıyla yalnızca toplam süre ölçümlenebilmektedir.
<i>All_Template</i>	Kontrolü yapılacak kurallar yine ilk senaryodaki gibi tek bir xslt dosyasında toplanmıştır ancak template kullanıldığı için fatura xml, yapısına göre bir arada bulunan kurallar birlikte ölçülecek şekilde toplam 7 ölçüm değeri elde edilmiştir.
<i>Single_Original</i>	Her kural ayrı bir xslt dosyası kullanılarak ölçülmüş olup, template kullanılmadığı için yalnızca her kural için toplam süre ölçümü yapılmıştır.
<i>Single_Template</i>	3. Senaryodaki gibi her kural ayrı bir xslt dosyasına yazılmış, olup template kullanımı sayesinde her kuralın süre ölçümü yapılabildiği gibi, aynı zamanda her kural için geçen toplam süre ölçümü de yapılmıştır.

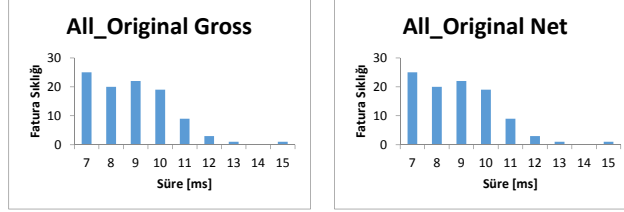
Tablo 1. Olası Test Senaryoları

Test için kullanılan bilgisayar 8GB RAM ve Intel Core i5-3230M işlemcisine sahip iken, performans ölçümleme aracı olarak Saxon XSLT ve XQuery Processor Home Edition Version 9.7 kullanılmıştır.

3.1 Test Senaryoları

Performans testlerinin yürütülmesi adına toplam dört farklı senaryo belirlenmiştir. Bu senaryolar Tablo 1’ de detaylı bir şekilde görülebilmektedir.

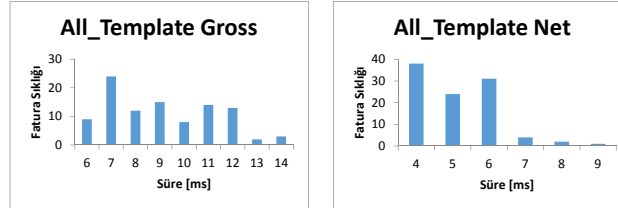
3.2 Testlerin Yürütülmesi



Şekil 2. All_Original senaryosuna ait *gross* ve *net* histogramlar

Senaryolar belirlendikten sonra, ölçüm sonuçlarının daha tutarlı olması ve hatalardan arındırılması için her senaryo ölçümü 100 kez tekrarlanarak sonuçlar kaydedilmiştir. Daha sonra ölçüm sonuçlarının yer aldığı HTML dosyalarını ayrıştırma amacıyla *jsoup* kütüphanesi kullanılarak bir parser yazılmıştır. Bu parser HTML dosyalarını okuyarak ölçüm sonuçlarını CSV formatına çevirmektedir. CSV formatına gelen data analiz edilerek bölüm 3.3’ de yorumlanarak sunulmuştur.

3.3 Test Sonuçları



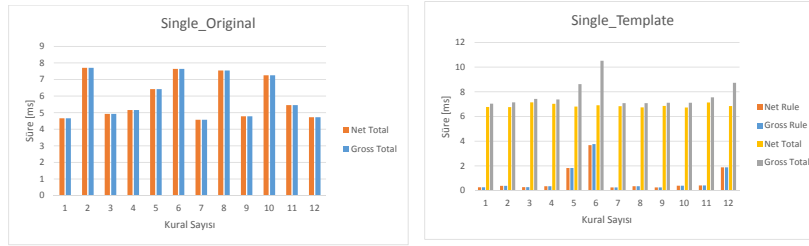
Şekil 3. All_Template senaryosuna ait *gross* ve *net* histogramlar

All_Original senaryosu için yapılan 100 ölçüm sonuçları *net* ve *gross* olarak histogram formatına getirilmiştir. Şekil 2’ de ilgili histogram görülmektedir.

Benzer şekilde *All_Template* senaryosu için yapılan 100 ölçüm sonuçları *net* ve *gross* olarak histogram formatına getirilmiştir. Şekil 3’ de ilgili histogram görülmektedir.

Beklenildiği gibi *All_Original* senaryosunda (bkz. Şekil 2) *gross* ve *net* süreler aynı çıkmaktadır. Buna karşın *All_Template* senaryosunda (bkz. Şekil 3) template yapısından dolayı yalnızca kural kontrolünün ne kadar sürdüğü *net*

olarak ölçülebilirken aynı zamanda komut verildikten itibaren sonuç dosyalarının üretimine kadar geçen süre de *gross* olarak kaydedilebilmektedir. *Gross* ortalamalarına bakıldığında *All_Original* senaryosunda 8,37ms ve *All_Template* senaryosunda 8,60ms süre elde edilmektedir. Yaklaşık 0,23ms'lik bu süre farkı her iki senaryonun da çok yakın sürelerde çalıştığını göstermektedir. Ayrıca *All_Template* senaryosunda (bkz. Şekil 3) *net* sürelerin *gross* sürelere bölümüyle elde edilen sonuç toplam sürenin yaklaşık %53,7'sinin kural kontrolüne harcadığını, geri kalan sürenin ise diğer işlemlere (overhead) harcadığını belirtmektedir.



(a) Single_Original

(b) Single_Template

Şekil 4. Single_Original ve _Template senaryosuna ait ölçümler

Şekil 4a' da üçüncü senaryo olan *Single_Original* senaryosunda ise analiz için histogram yerine her kural için ortalama *net* ve *gross* süreleri hesaplanmıştır. İlk senaryoda olduğu gibi (bkz. Şekil 2) template yapısının yokluğundan dolayı bu süreler birbirine eşit çıkmıştır. Bu ortalama süreler toplandığında bir faturanın bütün kurallardan bu yöntemle geçirilmesi esnasında toplam olarak yaklaşık 70,8ms harcandığı görülmektedir. *All_Original* senaryosunun (bkz. Şekil 2) ortalama süresi olan 8,37ms ile kıyaslandığında bu yöntemin daha yavaş olduğu görülmektedir.

Şekil 4b' de son senaryo olan *Single_Template* senaryosunda ise yine template yapısından faydalanılarak her kural kontrolü için geçen süreyle birlikte toplam süreler de hesaplanmıştır. Ortalama *Gross_Total* ve *Net_Total* süreleri sırasıyla 92,8ms ve 82,6ms olarak hesaplanmıştır. Bu durumda bu senaryonun *Single_Original* senaryosundan bkz. Şekil 4a) daha yavaş çalıştığı görülmektedir. Bu durumun sebebi ise template yapısının getirdiği ekstra iş yükü olmaktadır. Kural bazında bakıldığında ise kontrolün gerçekleştiği sürenin toplam süreye oranının 5. ,6. ve 12.kurallar hariç %10 seviyesinin altında olduğu gözlemlenmiştir.

4 Sonuç

Bu çalışma kapsamında UBL-TR formatında olan e-Fatura verisinin imza sonrası dört farklı kontrolden geçmesi gerektiği anlatılmaktadır. Yukarıda da yer yer bahsedildiği gibi bu dört kontrol sırasıyla aşağıdaki gibi verilmiştir:

- XSD: e-Fatura verisinin yapısal kontrolü
- Schematron: e-Fatura verisinin zorunlu (GİB tarafından belirlenen) anlamsal tutarlılık kontrolü
- Alıcı-Genel Kurallar: Bir vergi mükellefinin ürün, servis ya da hizmet alımında belirlediği genel kurallar
- Alıcı-Satıcı Özel Kurallar: Bir e-Fatura düzenlenmesinde alıcı satıcı konumunda bulunan vergi mükelleflerinin, ya da sadece alıcının belirlediği ilgili satıcıya dair özel kurallar

UBL-TR formatında bulunan, imzalı her bir e-Fatura verisi bu kontrollerden geçtiği için, bu kontrolleri kapsayan kuralların bir e-Fatura bazında işleyiş şekli e-Fatura'nın işleme süresinin önemli bir kısmını oluşturmaktadır. Bu çalışma kapsamında örnek bir UBL-TR formatındaki örnek bir e-Fatura XML dosyası ile 12 farklı kural içeren bir XSLT dosyası Saxon kütüphanesi kullanılarak XSD, Schematron ve alıcı genel kontrolleri gerçekleştirilmiştir. Bu işlem hem kuralların hepsi tek bir XSLT dosyası içerisindeyken hem de kurallar ayrıştırılıp 12 ayrı XSLT dosyası halindeyken yapılmıştır. Yapılan testlerin analizi sonucunda tek bir XSLT dosyasında bulunan kuralların her bir kural için farklı XSLT dosyalarına ayrılması sistemi yavaşlatmıştır. Yapılan testler sonucunda takribi olarak toplam sürenin yaklaşık %53,7'sinin kural kontrolüne harcadığını, geri kalan sürenin (%46,3'ü) ise diğer işlemlere (overhead) harcadığı gözlemlenmiştir. Hal böyle olunca, birbirinden bağımsız dört farklı XSLT dosyasının tek bir dosyaya indirgenmesi, sistem yükünü bariz bir şekilde hafifletecektir.

5 TEŞEKKÜR

Bu çalışma Ar-Ge Merkezi İdea Teknoloji Çözümleri Bilg. San.Tic. A. Ş. kurumu tarafından 3150333 no'lu, ELICIT: e-Fatura Uyumluluk ve Hile Denetim Aracı başlıklı TEYDEB-1501 projesi kapsamında desteklenmiştir.

6 Kaynakça

- [1] S. Bayar, M. G. Ülkar, A. Şen, "Kullanıcı Tarafında E-Belge Oluşturma ve Yazdırma Yazılım Deneyimleri", 9. Ulusal Yazılım Mühendisliği Sempozyumu (UYMS'15), İzmir, Türkiye, 09-11 Eylül 2015.
- [2] Ernst & Young, Son düzenlemeler ışığında Elektronik Fatura ve Elektronik Defter Uygulamaları, Temmuz 2013
- [3] Gelir İdaresi Başkanlığı 2015 Yılı Faaliyet Raporu. <http://www.gib.gov.tr/>
- [4] S. Bayar, M. G. Ülkar, U. Doğan, "Türkiye' de ve Avrupa'da E-Fatura Uygulaması", XVII. Akademik Bilişim Konferansı, AB 2015, 4 - 6 Şubat 2015, Anadolu Üniversitesi, Eskişehir.